Distributed Message Service for Kafka

User Guide

Issue	01
Date	2023-08-18

HUAWEI CLOUD COMPUTING TECHNOLOGIES CO., LTD.

Trademarks and Permissions

Notice

The purchased products, services and features are stipulated by the contract made between Huawei Cloud and the customer. All or part of the products, services and features described in this document may not be within the purchase scope or the usage scope. Unless otherwise specified in the contract, all statements, information, and recommendations in this document are provided "AS IS" without warranties, guarantees or representations of any kind, either express or implied.

The information in this document is subject to change without notice. Every effort has been made in the preparation of this document to ensure accuracy of the contents, but all statements, information, and recommendations in this document do not constitute a warranty of any kind, express or implied.

Contents

1 Permissions Management

1.1 Creating a User and Granting DMS for Kafka Permissions

This section describes how to use **Identity and Access Management (IAM)** for fine-grained permissions control for your Distributed Message Service (DMS) for Kafka resources. With IAM, you can:

- Create IAM users for personnel based on your enterprise's organizational structure. Each IAM user has their own identity credentials for accessing DMS for Kafka resources.
- Grant users only the permissions required to perform a given task based on their job responsibilities.
- Entrust another HUAWEI ID or cloud service to perform efficient O&M on your DMS for Kafka resources.

If your HUAWEI ID meets your permissions requirements, you can skip this section.

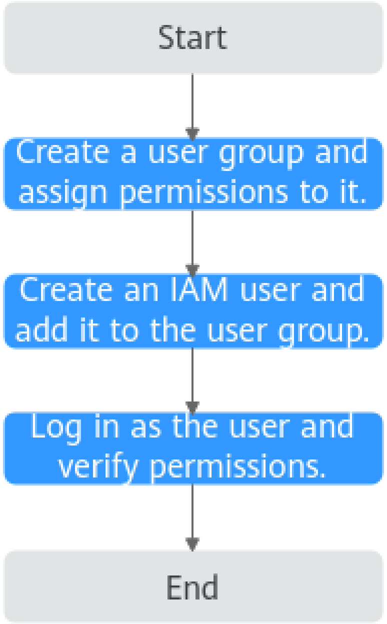
This section describes the procedure for granting permissions (see **Figure 1-1**).

Prerequisites

Learn about the permissions (see **System-defined roles and policies supported by DMS for Kafka**) supported by DMS for Kafka and choose policies according to your requirements. For the permissions of other services, see **System Permissions**.

Process Flow

Figure 1-1 Process for granting DMS for Kafka permissions


Start
Create a user group and assign permissions to it.
Create an IAM user and add it to the user group.
Log in as the user and verify permissions.
End

1. On the IAM console, **create a user group and grant it permissions**.

   DMS ReadOnlyAccess is used as an example.

2. **Create an IAM user and add it to the created user group**.
3. **Log in as the IAM user** and verify permissions.

   In the authorized region, perform the following operations:

   – Choose **Service List** > **Distributed Message Service (for Kafka)**. Then click **Buy Instance** on the console of DMS for Kafka. If a message appears indicating that you have insufficient permissions to perform the operation, the **DMS ReadOnlyAccess** policy is in effect.

   – Choose **Service List** > **Elastic Volume Service**. If a message appears indicating that you have insufficient permissions to access the service, the **DMS ReadOnlyAccess** policy is in effect.

1.2 DMS for Kafka Custom Policies

Custom policies can be created to supplement the system-defined policies of DMS for Kafka. For the actions that can be added for custom policies, see **Permissions Policies and Supported Actions**.

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.

For details, see **Creating a Custom Policy**. The following section contains examples of common DMS for Kafka custom policies.

NOTE

- DMS for Kafka permissions policies are based on DMS. Therefore, when assigning permissions, select DMS permissions policies.
- Due to data caching, a policy involving Object Storage Service (OBS) actions will take effect five minutes after it is attached to a user, user group, or project.

Example Custom Policies

- Example 1: Allowing users to delete and restart instances

```
{
    "Version": "1.1",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "dms:instance:modifyStatus",
                "dms:instance:delete"
            ]
        }
    ]
}
```

- Example 2: Denying instance deletion

  A policy with only "Deny" permissions must be used in conjunction with other policies to take effect. If the permissions assigned to a user include both "Allow" and "Deny", the "Deny" permissions take precedence over the "Allow" permissions.

  For example, if you want to assign all of the permissions of the **DMS FullAccess** policy to a user, except for deleting instances, you can create a custom policy to deny only instance deletion. When you apply both the **DMS FullAccess** policy and the custom policy denying instance deletion, since "Deny" always takes precedence over "Allow", the "Deny" will be applied for that one conflicting permission. The user will then be able to perform all operations on instances except deleting instances. The following is an example of a deny policy:

```
{
        "Version": "1.1",
        "Statement": [
                {
                        "Effect": "Deny",
                        "Action": [
                                "dms:instance:delete"
                        ]
                }
        ]
}
```

1.3 DMS for Kafka Resources

A resource is an object that exists within a service. DMS for Kafka resources are **kafka**. You can select them by specifying their paths.

Table 1-1 DMS for Kafka resources and their paths

Resource	Resource Name	Path
kafka	Instance	[Format] DMS:*:*: kafka: *instance ID* [Notes] For instance resources, IAM automatically generates the prefix (**DMS:*:*:kafka:**) of the resource path. For the path of a specific instance, add the *instance ID* to the end. You can also use an asterisk ***** to indicate any instance. For example: **DMS:*:*:kafka:*** indicates any Kafka instance.

1.4 DMS for Kafka Request Conditions

Request conditions are useful for fine tuning when a custom policy takes effect. A request condition consists of a condition key and operator. Condition keys are either global or service-level and are used in the Condition element of a policy statement. **Global condition keys** (starting with **g:**) are available for operations of all services, while service-level condition keys (starting with a service name such as *dms:*) are available only for operations of a specific service. An operator must be used together with a condition key to form a complete condition statement.

DMS for Kafka has a group of predefined condition keys that can be used in IAM. For example, to define an "Allow" permission, you can use the condition key **dms:ssl** to check whether SASL is enabled for a Kafka instance. The following table lists the predefined condition keys of DMS for Kafka.

Table 1-2 Predefined condition keys of DMS for Kafka

Condition Key	Operator	Description
dms:publicIP	Bool IsNullOrEmpty BoolIfExists	Whether public access is enabled
dms:ssl	Bool IsNullOrEmpty BoolIfExists	Whether SASL is enabled

2 Preparing Required Resources

Overview

Before creating a Kafka instance, ensure the availability of resources, including a virtual private cloud (VPC), subnet, security group, and security group rules. Each Kafka instance is deployed in a VPC and bound to a specific subnet and security group. In this way, Kafka provides an isolated virtual network environment and security protection policies that you can easily configure and manage.

To access a Kafka instance over a public network, prepare an elastic IP address (EIP) in advance.

Required Resources

Table 2-1 lists the resources required by a Kafka instance.

Table 2-1 Kafka resources

Resource	Requirement	Operations
VPC and subnet	Different Kafka instances can use the same or different VPCs and subnets based on site requirements. Note the following when creating a VPC and a subnet: ● The VPC must be created in the same region as the Kafka instance. ● Use the default settings when creating a VPC and subnet.	For details on how to create a VPC and subnet, see **Creating a VPC**. If you need to create and use a new subnet in an existing VPC, see **Creating a Subnet for the VPC**.

Resource	Requirement	Operations
Security group	Different Kafka instances can use the same or different security groups. Note the following when creating a security group: ● Set **Template** to **Custom**. ● To use Kafka instances, add the security group rules described in **Table 2-2**. Other rules can be added based on site requirements. **NOTE** After a security group is created, its default inbound rule allows communication among ECSs within the security group and its default outbound rule allows all outbound traffic. In this case, you can access a Kafka instance within a VPC, and do not need to add rules according to **Table 2-2**.	For details on how to create a security group, see **Creating a Security Group**. For details on how to add rules to a security group, see **Adding a Security Group Rule**.
EIP	Note the following when creating EIPs: ● The EIPs must be created in the same region as the Kafka instance. ● The number of EIPs must be the same as the number of Kafka instance brokers.	For details about how to create an EIP, see **Assigning an EIP**.

Table 2-2 Security group rules

Direction	Protocol	Port	Source	Description
Inbound	TCP	9094	0.0.0.0/0	Access a Kafka instance through the public network (without SSL encryption).
Inbound	TCP	9092	0.0.0.0/0	Access a Kafka instance within a VPC (without SSL encryption).
Inbound	TCP	9095	0.0.0.0/0	Access a Kafka instance through the public network (with SSL encryption).
Inbound	TCP	9093	0.0.0.0/0	Access a Kafka instance within a VPC (with SSL encryption).

Direction	Protocol	Port	Source	Description
Inbound	TCP	9999	0.0.0.0/0	Access Kafka Manager.
Inbound	TCP	9011	198.19.128.0/17	Access a Kafka instance across VPCs using a VPC endpoint (with or without SSL).
Inbound	TCP	9011	0.0.0.0/0	Access a Kafka instance using DNAT (with or without SSL).

3 Buying an Instance

Scenario

Kafka instances are physically isolated and exclusively occupied by each tenant. You can customize the computing capabilities and storage space of an instance based on service requirements.

Before You Start

- Before buying a Kafka instance, ensure that a VPC configured with security groups and subnets is available.
- (Optional) If you want to access a Kafka instance over a public network, prepare an elastic IP address (EIP) in advance.

Procedure

Step 1 Log in to the management console.

Step 2 Click 📍 in the upper left corner to select a region.

NOTE

Select the region your application is in.

Step 3 Click ☰ and choose **Middleware** > **Distributed Message Service for Kafka** to open the console of DMS for Kafka.

Step 4 Click **Buy Instance** in the upper right corner of the page.

By default, you can create a maximum of 100 Kafka instances for each project. To create more instances, contact customer service to increase your quota.

Step 5 Specify **Billing Mode**, **Region**, **Project**, and **AZ**.

Step 6 Enter an instance name and select an enterprise project.

Step 7 Configure the following instance parameters:

Specifications: Select **Default** or **Custom**.

If you select Default, specify the version, broker flavor, number of brokers, and storage space to be supported by the Kafka instance based on the site requirements.

1. **Version**: Kafka v1.1.0, v2.3.0, and v2.7 are supported. v2.7 is recommended. **The version cannot be changed once the instance is created.**
2. **CPU Architecture**: The x86 architecture is supported.
3. **Broker Flavor**: Select broker specifications that best fit your business needs.

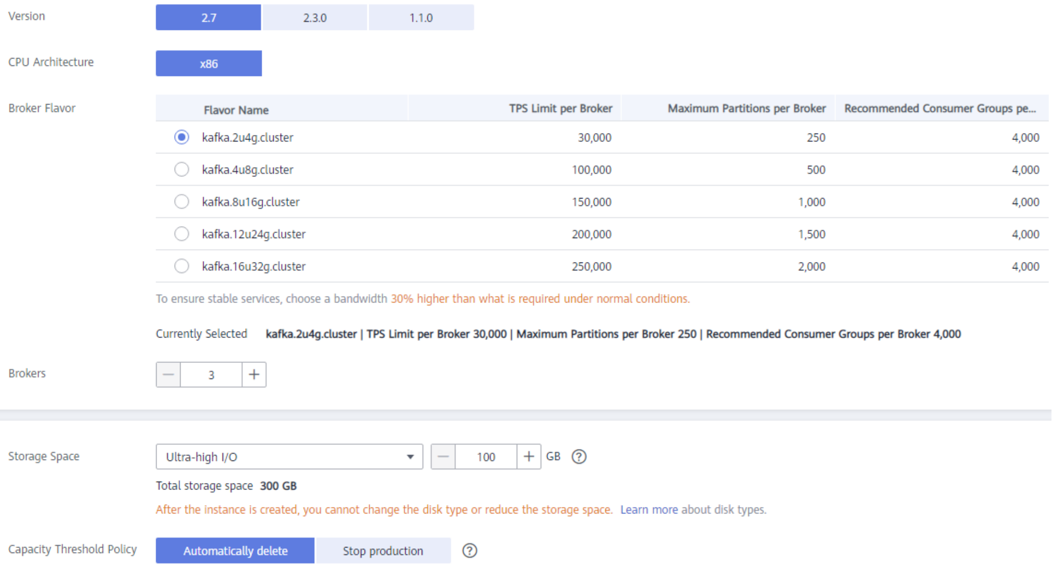
   Maximum number of partitions per broker x Number of brokers = Maximum number of partitions of an instance. If the total number of partitions of all topics exceeds the maximum number of partitions allowed for an instance, topic creation will fail.
4. For **Brokers**, specify the broker quantity.
5. **Storage Space**: Disk type and total disk space for storing the instance data. **The disk type cannot be changed once the instance is created.**

   The storage space is the total space to be consumed by all replicas. Specify the storage space based on the expected service message size and the number of replicas. For example, if the required disk size to store the data for the retention period is 100 GB, the disk capacity must be at least: 100 GB x Number of replicas + 100 GB (reserved).

   Disks are formatted when an instance is created. As a result, the actual available disk space is 93% to 95% of the total disk space.

   - Flavor **kafka.2u4g.cluster**: The value range of **Storage Space** is 300–300,000 GB.
   - Flavor **kafka.4u8g.cluster**: The value range of **Storage Space** is 300–600,000 GB.
   - Flavor **kafka.8u16g.cluster**: The value range of **Storage Space** is 300–900,000 GB.
   - Flavor **kafka.12u24g.cluster**: The value range of **Storage Space** is 300–900,000 GB.
   - Flavor **kafka.16u32g.cluster**: The value range of **Storage Space** is 300–900,000 GB.
6. **Capacity Threshold Policy**: policy used when the disk usage reaches the threshold. The capacity threshold is 95%.
   - **Automatically delete**: Messages can be created and retrieved, but 10% of the earliest messages will be deleted to ensure sufficient disk space. This policy is suitable for scenarios where no service interruption can be tolerated. Data may be lost.
   - **Stop production**: New messages cannot be created, but existing messages can still be retrieved. This policy is suitable for scenarios where no data loss can be tolerated.

Figure 3-1 Default specifications



Version: 2.7 | 2.3.0 | 1.1.0

CPU Architecture: x86

Broker Flavor

	Flavor Name	TPS Limit per Broker	Maximum Partitions per Broker	Recommended Consumer Groups pe...
●	kafka.2u4g.cluster	30,000	250	4,000
○	kafka.4u8g.cluster	100,000	500	4,000
○	kafka.8u16g.cluster	150,000	1,000	4,000
○	kafka.12u24g.cluster	200,000	1,500	4,000
○	kafka.16u32g.cluster	250,000	2,000	4,000

To ensure stable services, choose a bandwidth 30% higher than what is required under normal conditions.

Currently Selected kafka.2u4g.cluster | TPS Limit per Broker 30,000 | Maximum Partitions per Broker 250 | Recommended Consumer Groups per Broker 4,000

Brokers: 3

Storage Space: Ultra-high I/O, 100 GB

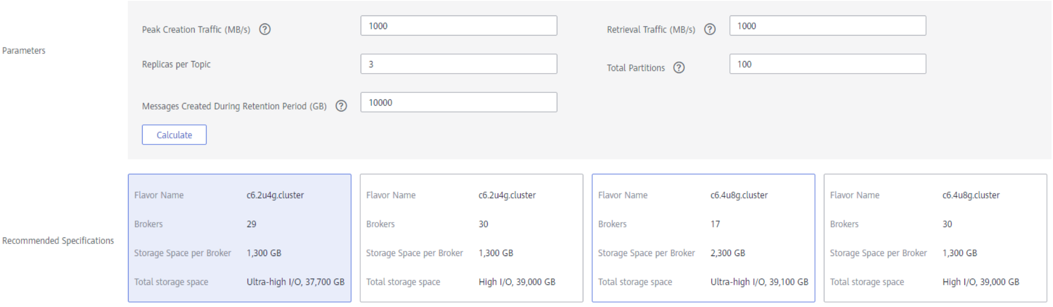
Total storage space 300 GB

After the instance is created, you cannot change the disk type or reduce the storage space. Learn more about disk types.

Capacity Threshold Policy: Automatically delete | Stop production

If you select Custom, the system calculates the number of brokers and broker storage space for different flavors based on your specified peak creation traffic, retrieval traffic, number of replicas per topic, total number of partitions, and size of messages created during the retention period. You can select one of the recommended flavors as required.

Figure 3-2 Specification calculation



Parameters

Peak Creation Traffic (MB/s): 1000

Retrieval Traffic (MB/s): 1000

Replicas per Topic: 3

Total Partitions: 100

Messages Created During Retention Period (GB): 10000

Recommended Specifications

Flavor Name	Brokers	Storage Space per Broker	Total storage space
c6.2u4g.cluster	29	1,300 GB	Ultra-high I/O, 37,700 GB
c6.2u4g.cluster	30	1,300 GB	High I/O, 39,000 GB
c6.4u8g.cluster	17	2,300 GB	Ultra-high I/O, 39,100 GB
c6.4u8g.cluster	30	1,300 GB	High I/O, 39,000 GB

Step 8 Configure the instance network parameters.

- Select a VPC and a subnet.

  A VPC provides an isolated virtual network for your Kafka instances. You can configure and manage the network as required.

  NOTE

  After the Kafka instance is created, its VPC and subnet cannot be changed.

- Select a security group.

  A security group is a set of rules for accessing a Kafka instance. You can click **Manage Security Group** to view or create security groups on the network console.

Step 9 Configure the username and password for logging in to Kafka Manager. **The Kafka Manager username cannot be changed once the instance is created.**

Kafka Manager is an open-source tool for managing Kafka clusters. After a Kafka instance is created, you can go to the instance details page to obtain the address for logging in to Kafka Manager. In Kafka Manager, you can view the monitoring statistics and broker information of your Kafka clusters.

Step 10 Click **Advanced Settings** to configure more parameters.

1. Configure public access.

   Public access is disabled by default. You can enable or disable it as required.

   After public access is enabled, configure an IPv4 EIP for each broker.

   After enabling **Public Access**, you can enable or disable **Intra-VPC Plaintext Access**. **If it is enabled, data will be transmitted in plaintext when you connect to the instance through a private network, regardless of whether SASL_SSL is enabled. This setting cannot be changed after the instance is created.** Exercise caution. If you want to use a different setting, you must create a new instance.

2. Configure **Kafka SASL_SSL**.

   This parameter indicates whether to enable SASL authentication when a client connects to the instance. If you enable **Kafka SASL_SSL**, data will be encrypted before transmission to enhance security.

   Kafka SASL_SSL is disabled by default. You can enable or disable it as required. **This setting cannot be changed after the instance is created.** If you want to use a different setting, you must create a new instance.

   If you enable **Kafka SASL_SSL**, you can determine whether to enable **SASL/PLAIN**. If **SASL/PLAIN** is disabled, the SCRAM-SHA-512 mechanism is used to transmit data. If **SASL/PLAIN** is enabled, both the SCRAM-SHA-512 and PLAIN mechanisms are supported. You can select either of them as required. The **SASL/PLAIN** setting cannot be changed once the instance is created.

   What are SCRAM-SHA-512 and PLAIN mechanisms?

   - SCRAM-SHA-512: uses the hash algorithm to generate credentials for usernames and passwords to verify identities. SCRAM-SHA-512 is more secure than PLAIN.
   - PLAIN: a simple username and password verification mechanism.

   If you enable **Kafka SASL_SSL**, you must also set the username and password for accessing the instance.

3. Configure **Automatic Topic Creation**.

   This setting is disabled by default. You can enable or disable it as required.

   If automatic topic creation is enabled, the system automatically creates a topic when a message is created in or retrieved from a topic that does not exist. This topic has the following default settings: 3 partitions, 3 replicas, aging time 72 hours, and synchronous replication and flushing disabled.

   After you change the value of the **log.retention.hours**, **default.replication.factor**, or **num.partitions** parameter, automatically created topics later use the new value. For example, if **num.partitions** is set to **5**, an automatically created topic will have the following settings: 5 partitions, 3 replicas, aging time 72 hours, and synchronous replication and flushing disabled.

4. Specify **Tags**.

Tags are used to identify cloud resources. When you have many cloud resources of the same type, you can use tags to classify them by dimension (for example, use, owner, or environment).

- If you have predefined tags, select a predefined pair of tag key and value. You can click **View predefined tags** to go to the Tag Management Service (TMS) console and view or create tags.
- You can also create new tags by specifying **Tag key** and **Tag value**.

Up to 20 tags can be added to each Kafka instance. For details about the requirements on tags, see **Managing Instance Tags**.

5. Enter a description of the instance.

Step 11 Click **Buy**.

Step 12 Confirm the instance information, read and agree to the *HUAWEI CLOUD Customer Agreement*, and click **Submit**.

Step 13 Return to the instance list and check whether the Kafka instance has been created.

It takes 3 to 15 minutes to create an instance. During this period, the instance status is **Creating**.

- If the instance is created successfully, its status changes to **Running**.
- If the instance fails to be created, view **Instance Creation Failures**. Delete the instance by referring to **Deleting an Instance** and create another instance. If the instance creation fails again, contact customer service.

NOTE

Instances that fail to be created do not occupy other resources.

----End

4 Accessing a Kafka Instance

4.1 Accessing a Kafka Instance Without SASL

This section describes how to use an open-source Kafka client to access a Kafka instance if SASL access is not enabled for the instance. There are two scenarios. For cross-VPC access, see **Cross-VPC Access to a Kafka Instance**. For DNAT-based access, see **Using DNAT to Access a Kafka Instance**.

For details on how to use Kafka clients in different languages, visit **https://cwiki.apache.org/confluence/display/KAFKA/Clients**.

NOTE

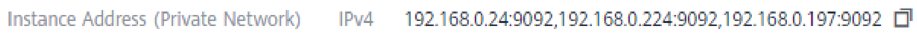
- The following describes the procedure for accessing a Kafka instance using CLI. To access an instance in your service code, see the **Developer Guide**.
- Each Kafka broker allows a maximum of 1000 connections from each IP address by default. Excess connections will be rejected. You can change the limit by referring to **Modifying Kafka Parameters**.

Prerequisites

- Security group rules have been correctly configured.

  To access a Kafka instance with SASL disabled, configure correct security group rules. For details about security group configuration requirements, see **Table 2-2**.
- The instance connection address has been obtained.
  - For intra-VPC access, use port 9092. Obtain the instance connection address in the **Connection** section of the **Basic Information** tab page.

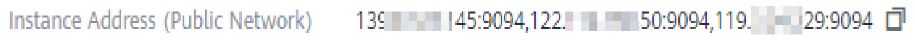
    Figure 4-1 Kafka instance connection addresses for intra-VPC access without SASL

    Instance Address (Private Network) IPv4 192.168.0.24:9092,192.168.0.224:9092,192.168.0.197:9092

  - For public access, use port 9094. Obtain the instance connection address in the **Connection** section of the **Basic Information** tab page.

Figure 4-2 Kafka instance connection addresses for public access without SASL



Instance Address (Public Network) 13[redacted]145:9094,122.[redacted]50:9094,119.[redacted]29:9094

- If automatic topic creation is not enabled for the Kafka instance, **create a topic** before connecting to the instance.
- Kafka CLI **v1.1.0**, **v2.3.0**, or **v2.7.2** is available. Ensure that the Kafka instance and the CLI are of the same version.
- An ECS has been created. For intra-VPC access, ensure that its VPC, subnet, and security group configurations are the same as those of the Kafka instance. **JDK v1.8.111 or later** has been installed on the ECS, and the **JAVA_HOME** and **PATH** environment variables have been configured as follows:

  Add the following lines to the **.bash_profile** file in the home directory as an authorized user. In this command, **/opt/java/jdk1.8.0_151** is the JDK installation path. Change it to the path where you install JDK.

  ```
  export JAVA_HOME=/opt/java/jdk1.8.0_151
  export PATH=$JAVA_HOME/bin:$PATH
  ```

  Run the **source .bash_profile** command for the modification to take effect.

Accessing the Instance Using CLI

The following uses Linux as an example.

Step 1 Decompress the Kafka CLI package.

Access the directory where the CLI package is stored and run the following command to decompress the package:

tar -zxf *[kafka_tar]*

In the preceding command, *[kafka_tar]* indicates the name of the CLI package.

For example:

tar -zxf kafka_2.12-2.7.2.tgz

Step 2 Access the **/bin** directory of the Kafka CLI.

In Windows, you need to access the **/bin/windows** directory.

Step 3 Run the following command to create messages:

```
./kafka-console-producer.sh --broker-list ${connection-address} --topic ${topic-name}
```

Parameter description:

- *{connection-address}*: the address obtained in **Prerequisites**. For public access, use **Instance Address (Public Network)**. For intra-VPC access, use **Instance Address (Private Network)**.
- *{topic-name}*: the name of the topic created for the Kafka instance. If automatic topic creation has enabled for the Kafka instance, set this parameter to the name of a created topic or a topic that has not been created.

The following example uses connection addresses **10.3.196.45:9094,10.78.42.127:9094,10.4.49.103:9094**. After running the preceding command, you can send a message to the Kafka instance by writing it and pressing **Enter**. Each line of content is sent as a message.

```
[root@ecs-kafka bin]# ./kafka-console-producer.sh --broker-list
10.3.196.45:9094,10.78.42.127:9094,10.4.49.103:9094  --topic topic-demo
>Hello
>DMS
>Kafka!
>^C[root@ecs-kafka bin]#
```

To stop creating messages, press **Ctrl**+**C** to exit.

Step 4 Run the following command to retrieve messages:

```
./kafka-console-consumer.sh --bootstrap-server ${connection-address} --topic ${topic-name} --group $
{consumer-group-name} --from-beginning
```

Parameter description:

- *{connection-address}*: the address obtained in **Prerequisites**. For public access, use **Instance Address (Public Network)**. For intra-VPC access, use **Instance Address (Private Network)**.
- *{topic-name}*: the name of the topic created for the Kafka instance
- *{consumer-group-name}*: the consumer group name set based on your service requirements. **If a consumer group name has been specified in the configuration file, ensure that you use the same name in the command line. Otherwise, consumption may fail.** If a consumer group name starts with a special character, such as a number sign (#), the monitoring data cannot be displayed.

Example:

```
[root@ecs-kafka bin]#  ./kafka-console-consumer.sh --bootstrap-server
10.3.196.45:9094,10.78.42.127:9094,10.4.49.103:9094 --topic topic-demo --group order-test --from-beginning
Kafka!
DMS
Hello
^CProcessed a total of 3 messages
[root@ecs-kafka bin]#
```

To stop retrieving messages, press **Ctrl**+**C** to exit.

----End

4.2 Accessing a Kafka Instance with SASL

If you enable SASL_SSL when creating an instance, data will be encrypted before transmission for enhanced security.

For security purposes, **TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256** is supported.

This section describes how to use an open-source Kafka client to access a Kafka instance if SASL has been enabled for the instance. There are two scenarios. For cross-VPC access, see **Cross-VPC Access to a Kafka Instance**. For DNAT-based access, see **Using DNAT to Access a Kafka Instance**.

NOTE

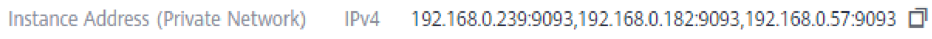
- If intra-VPC plaintext access is enabled for an instance, data is transmitted in plaintext when you connect to the instance through a private network. For details about how to connect, see **Accessing a Kafka Instance Without SASL**.
- Each Kafka broker allows a maximum of 1000 connections from each IP address by default. Excess connections will be rejected. You can change the limit by referring to **Modifying Kafka Parameters**.

Prerequisites

- Security group rules have been correctly configured.

  To access a Kafka instance with SASL enabled, configure correct security group rules. For details about security group configuration requirements, see **Table 2-2**.
- The instance connection address has been obtained.
  - For intra-VPC access, use port 9093. Obtain the instance connection address in the **Connection** section of the **Basic Information** tab page.

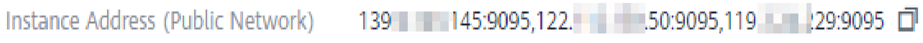
    Figure 4-3 Kafka instance connection addresses for intra-VPC access with SASL

    Instance Address (Private Network) IPv4 192.168.0.239:9093,192.168.0.182:9093,192.168.0.57:9093

  - For public access, use port 9095. Obtain the instance connection address in the **Connection** section of the **Basic Information** tab page.

    Figure 4-4 Kafka instance connection addresses for public access with SASL

    Instance Address (Public Network) 139[redacted]145:9095,122.[redacted].50:9095,119[redacted]29:9095

- The SASL mechanism in use is known.

  In the **Connection** area on the Kafka instance details page, view **SASL Mechanism**. If both SCRAM-SHA-512 and PLAIN are enabled, configure either of them for connections. If **SASL Mechanism** is not displayed, PLAIN is used by default.

  Figure 4-5 SASL mechanism in use

  SASL Mechanism SCRAM-SHA-512

- If automatic topic creation is not enabled for the Kafka instance, **create a topic** before connecting to the instance.
- The **client.truststore.jks** certificate has been downloaded. Click the Kafka instance to go to the **Basic Information** tab page. Click **Download** next to **SSL Certificate** in the **Connection** area. Download and decompress the package to obtain the client certificate file **client.truststore.jks**.
- Kafka CLI **v1.1.0**, **v2.3.0**, or **v2.7.2** is available. Ensure that the Kafka instance and the CLI are of the same version.

- An ECS has been created. For intra-VPC access, ensure that its VPC, subnet, and security group configurations are the same as those of the Kafka instance. JDK v1.8.111 or later has been installed on the ECS, and the **JAVA_HOME** and **PATH** environment variables have been configured as follows:

  Add the following lines to the **.bash_profile** file in the home directory as an authorized user. In this command, **/opt/java/jdk1.8.0_151** is the JDK installation path. Change it to the path where you install JDK.

  ```
  export JAVA_HOME=/opt/java/jdk1.8.0_151
  export PATH=$JAVA_HOME/bin:$PATH
  ```

  Run the **source .bash_profile** command for the modification to take effect.

Accessing the Instance Using CLI

The following uses Linux as an example.

Step 1 Map hosts to IP addresses in the **/etc/hosts** file on the host where the client is located, so that the client can quickly parse the instance brokers.

Set IP addresses to the instance connection addresses obtained in **Prerequisites**. Set hosts to the names of instance hosts. Specify a unique name for each host.

For example:

10.154.48.120 server01

10.154.48.121 server02

10.154.48.122 server03

Step 2 Decompress the Kafka CLI package.

Access the directory where the CLI package is stored and run the following command to decompress the package:

tar -zxf *[kafka_tar]*

In the preceding command, *[kafka_tar]* indicates the name of the CLI package.

For example:

tar -zxf kafka_2.12-2.7.2.tgz

Step 3 Modify the Kafka CLI configuration file based on the **SASL mechanism**.

- **If PLAIN is used**, find the **consumer.properties** and **producer.properties** files in the **/config** directory of the Kafka CLI and add the following content to the files:

  ```
  sasl.jaas.config=org.apache.kafka.common.security.plain.PlainLoginModule required \
  username="**********" \
  password="**********";
  sasl.mechanism=PLAIN

  security.protocol=SASL_SSL
  ssl.truststore.location={ssl_truststore_path}
  ssl.truststore.password=dms@kafka
  ssl.endpoint.identification.algorithm=
  ```

  Parameter description:

  - **username** and **password**: username and password you set when enabling SASL_SSL during Kafka instance creation or when creating a SASL_SSL user.
  - **ssl.truststore.location**: path for storing the **client.truststore.jks** certificate. Even in Windows, you need to use slashes (/) for the certificate path. Do not use backslashes (\), which are used by default for paths in Windows. Otherwise, the client will fail to obtain the certificate.
  - **ssl.truststore.password**: server certificate password, which must be set to **dms@kafka** and cannot be changed.
  - **ssl.endpoint.identification.algorithm**: whether to verify the certificate domain name. **This parameter must be left blank, which indicates disabling domain name verification**.
- **If SCRAM-SHA-512 is used**, find the **consumer.properties** and **producer.properties** files in the **/config** directory of the Kafka CLI and add the following content to the files:

```
sasl.jaas.config=org.apache.kafka.common.security.scram.ScramLoginModule required \
username="**********" \
password="**********";
sasl.mechanism=SCRAM-SHA-512

security.protocol=SASL_SSL
ssl.truststore.location={ssl_truststore_path}
ssl.truststore.password=dms@kafka
ssl.endpoint.identification.algorithm=
```

  Parameter description:

  - **username** and **password**: username and password you set when enabling SASL_SSL during Kafka instance creation or when creating a SASL_SSL user.
  - **ssl.truststore.location**: path for storing the **client.truststore.jks** certificate. Even in Windows, you need to use slashes (/) for the certificate path. Do not use backslashes (\), which are used by default for paths in Windows. Otherwise, the client will fail to obtain the certificate.
  - **ssl.truststore.password**: server certificate password, which must be set to **dms@kafka** and cannot be changed.
  - **ssl.endpoint.identification.algorithm**: whether to verify the certificate domain name. **This parameter must be left blank, which indicates disabling domain name verification**.

Step 4 Access the **/bin** directory of the Kafka CLI.

In Windows, you need to access the **/bin/windows** directory.

Step 5 Run the following command to create messages:

```
./kafka-console-producer.sh --broker-list ${connection-address} --topic ${topic-name} --producer.config ../
config/producer.properties
```

Parameter description:

- *{connection-address}*: the address obtained in **Prerequisites**. For public access, use **Instance Address (Public Network)**. For intra-VPC access, use **Instance Address (Private Network)**.
- *{topic-name}*: the name of the topic created for the Kafka instance If automatic topic creation has enabled for the Kafka instance, set this parameter to the name of a created topic or a topic that has not been created.

The following example uses connection addresses **10.3.196.45:9095,10.78.42.127:9095,10.4.49.103:9095**.

After running the preceding command, you can send a message to the Kafka instance by writing it and pressing **Enter**. Each line of content is sent as a message.

```
[root@ecs-kafka bin]#./kafka-console-producer.sh --broker-list
10.3.196.45:9095,10.78.42.127:9095,10.4.49.103:9095  --topic topic-demo --producer.config ../config/
producer.properties
>Hello
>DMS
>Kafka!
>^C[root@ecs-kafka bin]#
```

To stop creating messages, press **Ctrl**+**C** to exit.

Step 6 Run the following command to retrieve messages:

```
./kafka-console-consumer.sh --bootstrap-server ${connection-address} --topic ${topic-name} --group $
{consumer-group-name} --from-beginning  --consumer.config ../config/consumer.properties
```

Parameter description:

- *{connection-address}*: the address obtained in **Prerequisites**. For public access, use **Instance Address (Public Network)**. For intra-VPC access, use **Instance Address (Private Network)**.
- *{topic-name}*: the name of the topic created for the Kafka instance
- *{consumer-group-name}*: the consumer group name set based on your service requirements. **If a consumer group name has been specified in the configuration file, ensure that you use the same name in the command line. Otherwise, consumption may fail.** If a consumer group name starts with a special character, such as a number sign (#), the monitoring data cannot be displayed.

Example:

```
[root@ecs-kafka bin]#  ./kafka-console-consumer.sh --bootstrap-server
10.3.196.45:9095,10.78.42.127:9095,10.4.49.103:9095 --topic topic-demo --group order-test --from-
beginning --consumer.config ../config/consumer.properties
Hello
DMS
Kafka!
^CProcessed a total of 3 messages
[root@ecs-kafka bin]#
```

To stop retrieving messages, press **Ctrl**+**C** to exit.

----End

4.3 Kafka Manager

Kafka Manager is an open-source tool for managing Kafka. It can be used only through a web browser. In Kafka Manager, you can view the monitoring statistics and broker information of your Kafka clusters.

Prerequisites

Security group rules have been configured by referring to **Table 4-1**.

Table 4-1 Security group rule

Direction	Protocol	Port	Source	Description
Inbound	TCP	9999	0.0.0.0/0	Access Kafka Manager.

Logging In to Kafka Manager

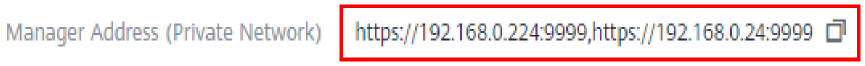
Step 1 Create a Windows ECS with the same VPC and security group configurations as the Kafka instance. For details, see **Purchasing an ECS**.

If public access has been enabled, this step is optional. You can access the instance using the local browser. You do not need to create a Windows ECS.

Step 2 Obtain the Kafka Manager address on the instance details page.

- If public network access has been disabled, the Kafka Manager address is **Manager Address (Private Network)**.

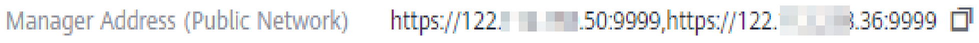
Figure 4-6 Kafka Manager address (private network)



Manager Address (Private Network) https://192.168.0.224:9999,https://192.168.0.24:9999

- If public network access has been enabled, the Kafka Manager address is **Manager Address (Public Network)**.

Figure 4-7 Kafka Manager address (public network)



Manager Address (Public Network) https://122.[redacted].50:9999,https://122.[redacted]3.36:9999

Step 3 Enter the Kafka Manager address in the web browser in the Windows ECS.

If public access is enabled, enter the Kafka Manager address in the address bar of the browser on the local PC. If public access is not enabled, log in to the ECS prepared in **Step 1** and enter the Kafka Manager address in the address bar of the browser on the ECS.

Step 4 Enter the username and password for logging in to Kafka Manager, which you set when creating the instance.

----End

Viewing Information in Kafka Manager

In Kafka Manager, you can view the monitoring statistics and broker information of your Kafka clusters.

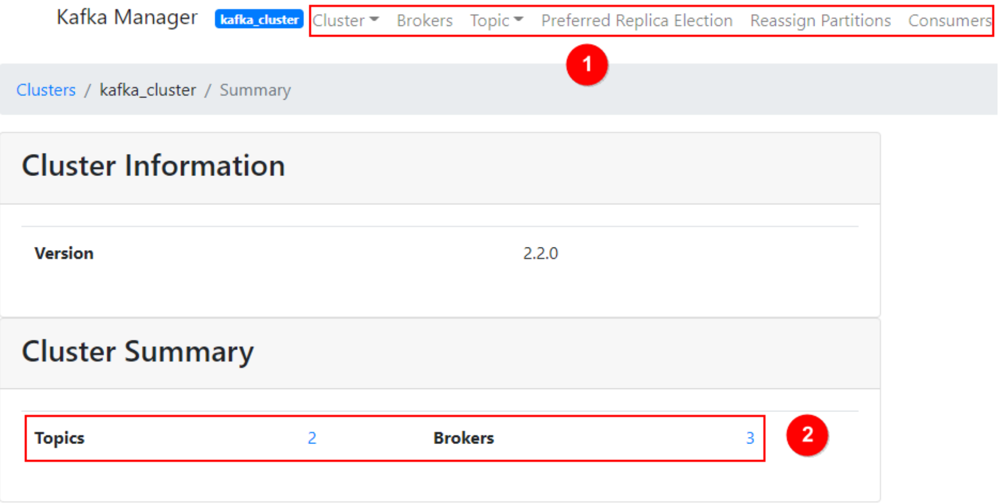
- Information about clusters

  Click **Clusters** to view the information about clusters. **Figure 4-8** shows an example of the cluster information.

  – The top navigation bar provides the following functions, as shown in the red box 1 in the figure.

        - **Cluster**: viewing the list of clusters and cluster information.
        - **Brokers**: viewing information about brokers of a cluster.
        - **Topic**: viewing information about topics in a cluster.
        - **Preferred Replica Election**: electing the leader (preferred replica) of a topic. This operation is not recommended.
        - **Reassign Partitions**: reassigning partitions. This operation is not recommended.
        - **Consumers**: viewing the status of consumer groups in a cluster.
    - Red box 2 shows an example of the cluster information summary, including the number of topics and brokers in the cluster.

Figure 4-8 Information about clusters



Cluster Information

Version	2.2.0

Cluster Summary

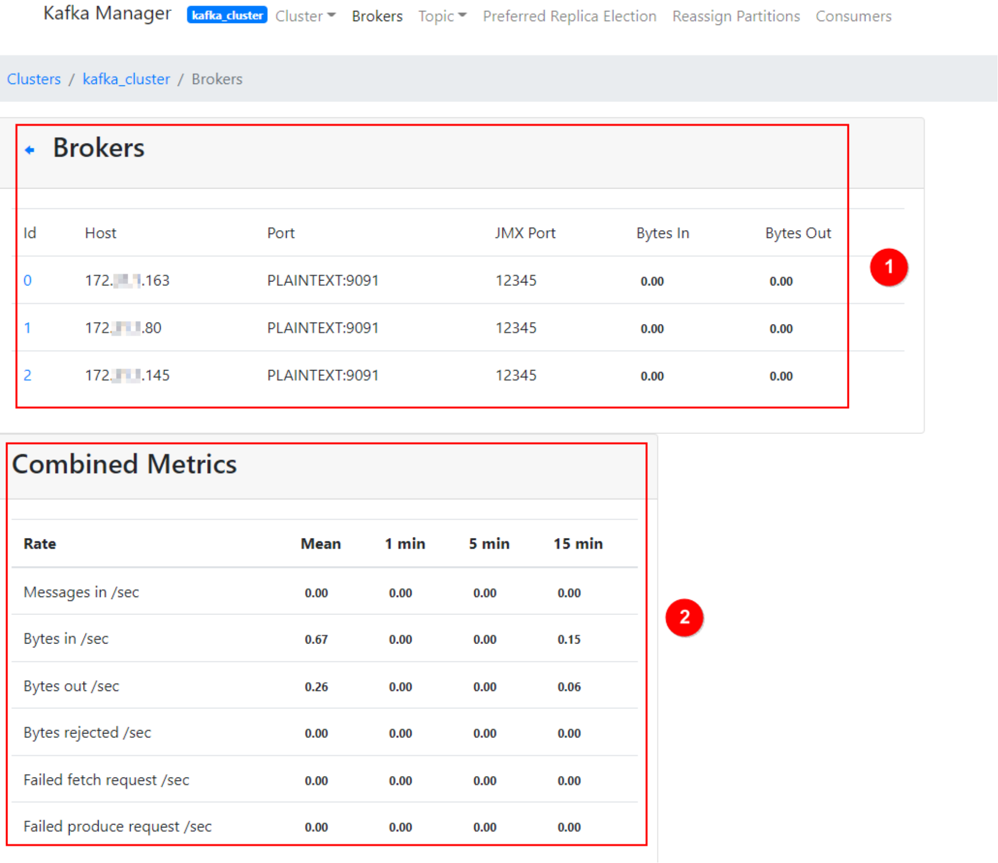
Topics	2	Brokers	3

- Combined information about all brokers of a cluster

  This page shows statistics of brokers of a cluster. **Figure 4-9** shows an example of the storage configuration.

    - Red box 1 shows the list of brokers, including number of incoming and outgoing bytes of different brokers.
    - Red box 2 shows the monitoring metrics of the cluster.

Figure 4-9 Viewing the combined information about all brokers in a cluster



Brokers

Id	Host	Port	JMX Port	Bytes In	Bytes Out
0	172.[redacted].163	PLAINTEXT:9091	12345	0.00	0.00
1	172.[redacted].80	PLAINTEXT:9091	12345	0.00	0.00
2	172.[redacted].145	PLAINTEXT:9091	12345	0.00	0.00

Combined Metrics

Rate	Mean	1 min	5 min	15 min
Messages in /sec	0.00	0.00	0.00	0.00
Bytes in /sec	0.67	0.00	0.00	0.15
Bytes out /sec	0.26	0.00	0.00	0.06
Bytes rejected /sec	0.00	0.00	0.00	0.00
Failed fetch request /sec	0.00	0.00	0.00	0.00
Failed produce request /sec	0.00	0.00	0.00	0.00

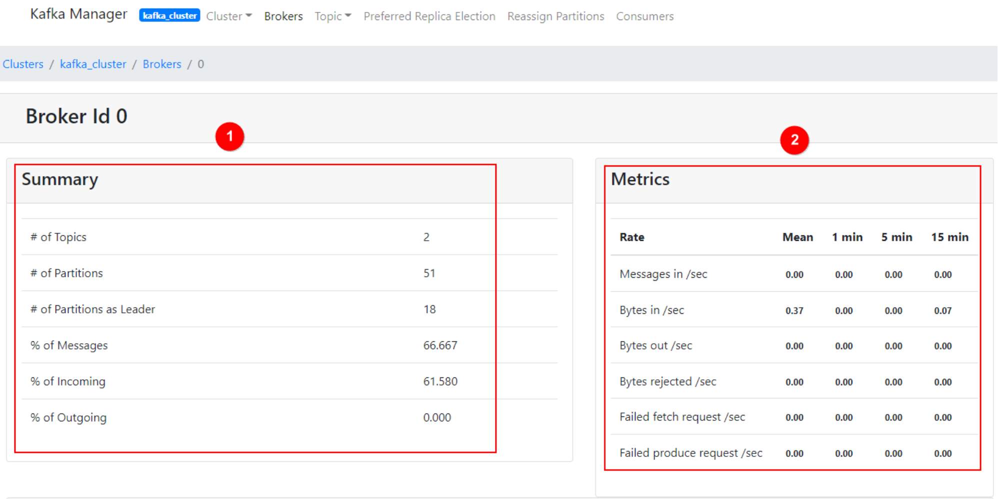
- Information about a specific broker

  Click the ID of a broker to view its statistics. **Figure 4-10** shows an example of the storage configuration.

  – Red box 1 shows the statistics of the broker, including the numbers of topics, partitions, and leaders, and percentages of messages, incoming traffic, and outgoing traffic.

  – Red box 2 shows the monitoring metrics of the broker.

Figure 4-10 Viewing information about a broker



Broker Id 0

Summary

# of Topics	2
# of Partitions	51
# of Partitions as Leader	18
% of Messages	66.667
% of Incoming	61.580
% of Outgoing	0.000

Metrics

Rate	Mean	1 min	5 min	15 min
Messages in /sec	0.00	0.00	0.00	0.00
Bytes in /sec	0.37	0.00	0.00	0.07
Bytes out /sec	0.00	0.00	0.00	0.00
Bytes rejected /sec	0.00	0.00	0.00	0.00
Failed fetch request /sec	0.00	0.00	0.00	0.00
Failed produce request /sec	0.00	0.00	0.00	0.00

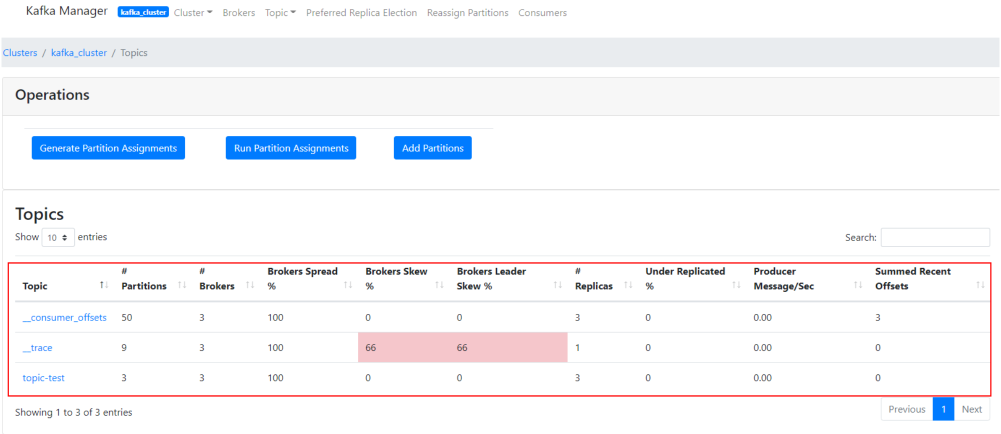
- Topics of an instance

  In the navigation bar, choose **Topic** > **List**. The displayed page shows the list of topics and information about the topics, as shown in **Figure 4-11**.

  NOTICE

  Topics starting with "__" are internal topics. To avoid service faults, do not perform any operation on these topics.

  Figure 4-11 Topics of an instance



Topics

Show 10 entries

Topic	# Partitions	# Brokers	Brokers Spread %	Brokers Skew %	Brokers Leader Skew %	# Replicas	Under Replicated %	Producer Message/Sec	Summed Recent Offsets
__consumer_offsets	50	3	100	0	0	3	0	0.00	3
__trace	9	3	100	66	66	1	0	0.00	0
topic-test	3	3	100	0	0	3	0	0.00	0

Showing 1 to 3 of 3 entries

- Details of a topic

  Click the name of a topic to view its details on the displayed page, as shown in **Figure 4-12**.

  – Red box 1: basic information about the topic, including **Replication**, **Number of Partitions**, and **Sum of Partition Offsets**.

  – Red box 2: information about partitions of different brokers.

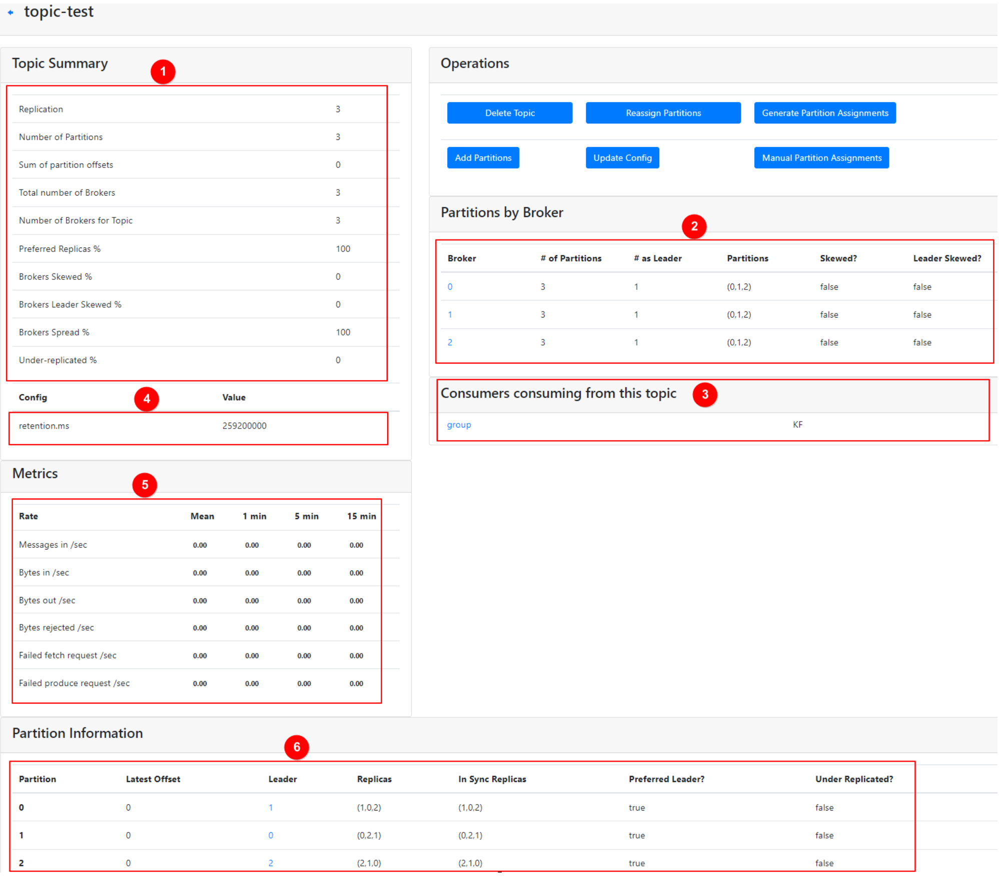
  – Red box 3: consumer groups of the topic. Click the name of a consumer group name to view its details.

  – Red box 4: configurations of the topic. For details, see **https://kafka.apache.org/documentation/#topicconfigs**.

  – Red box 5: monitoring metrics of the topic.

  – Red box 6: information about partitions in the topic, including **Latest Offset**, **Leader** of a partition, **Replicas**, and **In Sync Replicas**.

Figure 4-12 Details of a topic



topic-test

Topic Summary

Replication	3
Number of Partitions	3
Sum of partition offsets	0
Total number of Brokers	3
Number of Brokers for Topic	3
Preferred Replicas %	100
Brokers Skewed %	0
Brokers Leader Skewed %	0
Brokers Spread %	100
Under-replicated %	0

Config	Value
retention.ms	259200000

Operations

Partitions by Broker

Broker	# of Partitions	# as Leader	Partitions	Skewed?	Leader Skewed?
0	3	1	(0,1,2)	false	false
1	3	1	(0,1,2)	false	false
2	3	1	(0,1,2)	false	false

Consumers consuming from this topic

group	KF

Metrics

Rate	Mean	1 min	5 min	15 min
Messages in /sec	0.00	0.00	0.00	0.00
Bytes in /sec	0.00	0.00	0.00	0.00
Bytes out /sec	0.00	0.00	0.00	0.00
Bytes rejected /sec	0.00	0.00	0.00	0.00
Failed fetch request /sec	0.00	0.00	0.00	0.00
Failed produce request /sec	0.00	0.00	0.00	0.00

Partition Information

Partition	Latest Offset	Leader	Replicas	In Sync Replicas	Preferred Leader?	Under Replicated?
0	0	1	(1,0,2)	(1,0,2)	true	false
1	0	0	(0,2,1)	(0,2,1)	true	false
2	0	2	(2,1,0)	(2,1,0)	true	false

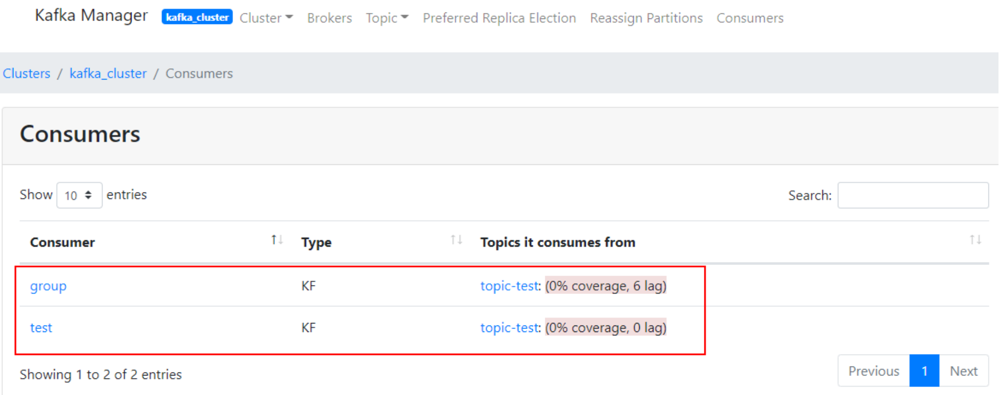
- List of consumers

  Click **Consumers** to view the list of consumers in a cluster.

  NOTE

  Only consumer groups that have retrieved messages in the last 14 days are displayed.

  Figure 4-13 Viewing the list of consumers

  Consumers

  Show 10 entries

  Search:

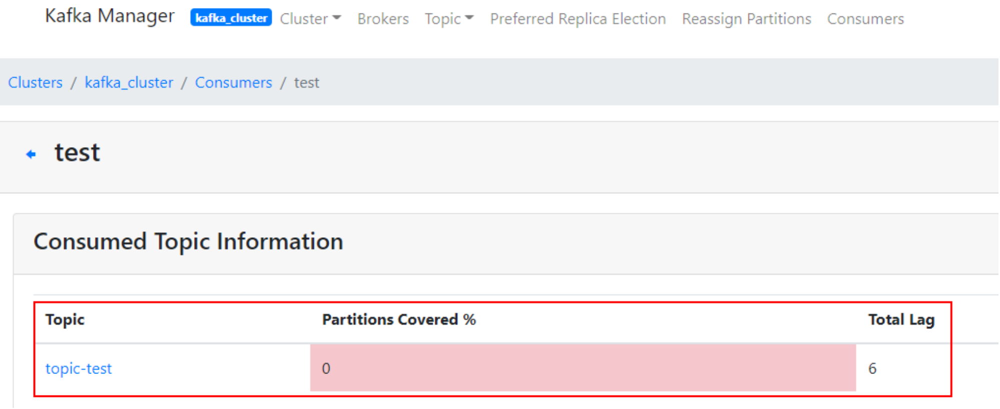
Consumer	Type	Topics it consumes from
group	KF	topic-test: (0% coverage, 6 lag)
test	KF	topic-test: (0% coverage, 0 lag)

  Showing 1 to 2 of 2 entries

- Details of a specific consumer

  Click the name of a consumer to view its details, including the list of topics in the consumer and the number of messages that can be retrieved in each topic (**Total Lag**).

Figure 4-14 Viewing consumer details



test

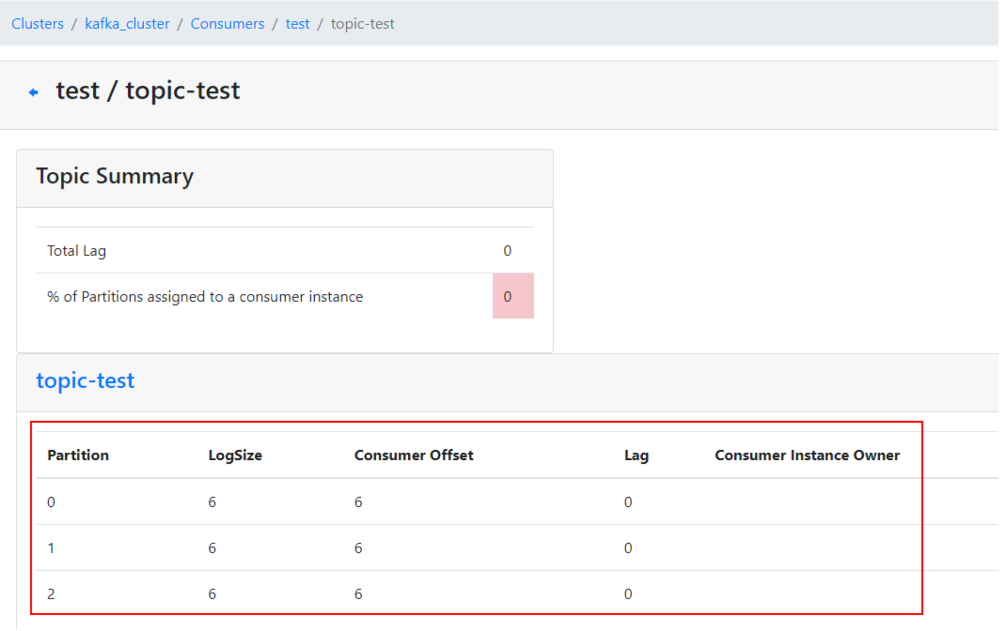
Consumed Topic Information

Topic	Partitions Covered %	Total Lag
topic-test	0	6

- Details of topics in a consumer

  Click the name of a topic to view retrieval details of different partitions in the topic, including **Partition**, the number of messages in a partition (**LogSize**), progress of the retrieval (**Consumer Offset**), number of remaining messages in the partition that can be retrieved (**Lag**), and the latest consumer that retrieved from the partition (**Consumer Instance Owner**).

Figure 4-15 Viewing details of a topic



test / topic-test

Topic Summary

Total Lag	0
% of Partitions assigned to a consumer instance	0

topic-test

Partition	LogSize	Consumer Offset	Lag	Consumer Instance Owner
0	6	6	0	
1	6	6	0	
2	6	6	0	

4.4 Cross-VPC Access to a Kafka Instance

Context

VPCs are logically isolated from each other. If a Kafka instance and a Kafka client are in different VPCs within a region, they cannot communicate with each other. In this case, you can use one of the following methods to access a Kafka instance across VPCs:

- Establish a VPC peering connection to allow two VPCs to communicate with each other. For details, see **VPC Peering Connection**.
- Use VPC Endpoint (VPCEP) to establish a cross-VPC connection.

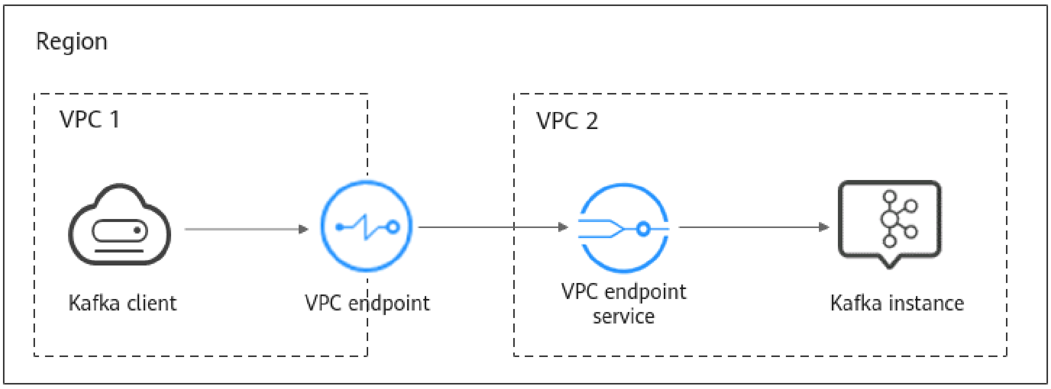
Scenario

The following describes how to use VPCEP to implement cross-VPC access.

VPCEP provides two types of resources: VPC endpoint services and VPC endpoints.

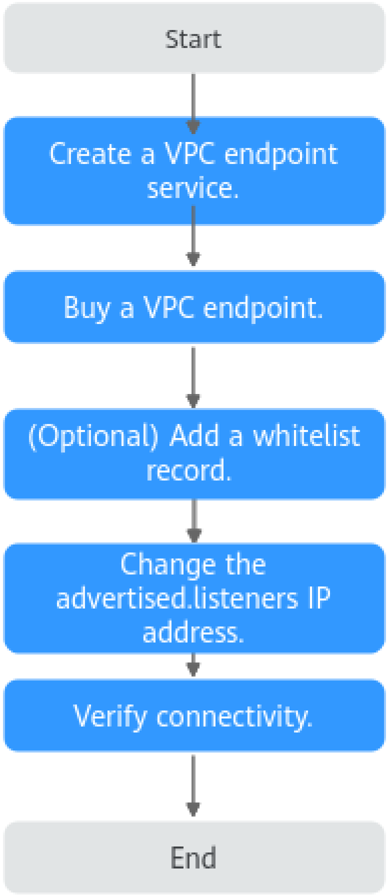
- A VPC endpoint service can be a Kafka instance which is accessed using VPC endpoints.
- A VPC endpoint is a secure and private channel for connecting a VPC to a VPC endpoint service.

Figure 4-16 Working principle of accessing a Kafka instance across VPCs



Procedure

Figure 4-17 Process for accessing a Kafka instance across VPCs


Start
Create a VPC endpoint service.
Buy a VPC endpoint.
(Optional) Add a whitelist record.
Change the advertised.listeners IP address.
Verify connectivity.
End

Creating a VPC Endpoint Service

Step 1 Log in to the management console.

Step 2 Click 📍 in the upper left corner to select a region.

NOTE

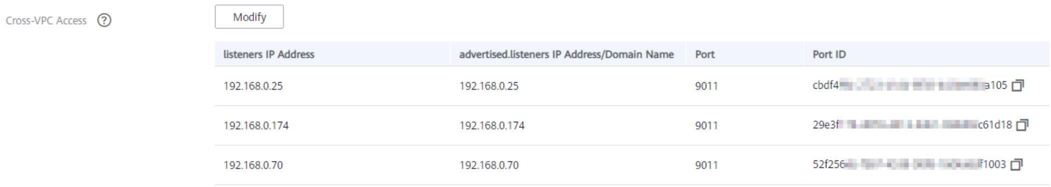
Select the region where your Kafka instance is located.

Step 3 Click ☰ and choose **Application** > **Distributed Message Service (for Kafka)** to open the console of DMS for Kafka.

Step 4 Click the desired Kafka instance to view the instance details.

Step 5 In the **Advanced Settings** section on the **Basic Information** tab page, obtain the listeners IP addresses and port IDs of the instance for **Cross-VPC Access**.

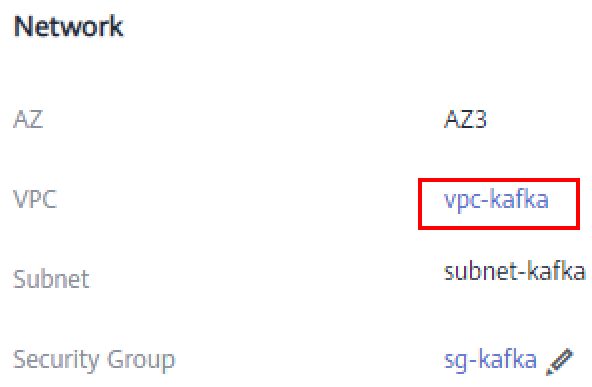
Figure 4-18 Cross-VPC access–related listeners IP addresses and corresponding port IDs of the Kafka instance



listeners IP Address	advertised.listeners IP Address/Domain Name	Port	Port ID
192.168.0.25	192.168.0.25	9011	cbdf4[redacted]a105
192.168.0.174	192.168.0.174	9011	29e3f[redacted]c61d18
192.168.0.70	192.168.0.70	9011	52f256[redacted]f1003

Step 6 In the **Network** section on the **Basic Information** tab page, view the VPC to which the Kafka instance belongs.

Figure 4-19 Viewing the VPC to which the Kafka instance belongs

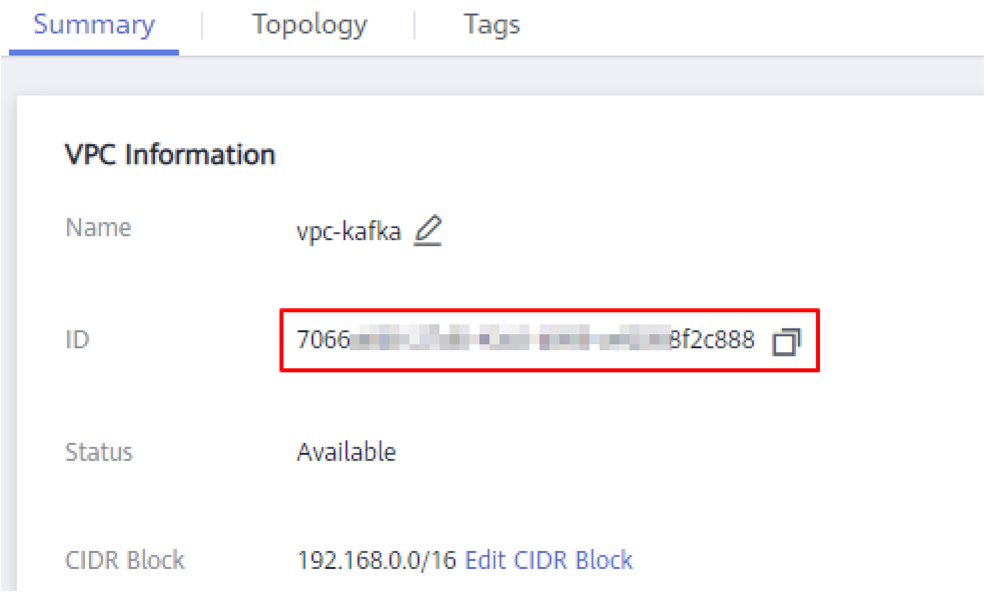


Network

AZ	AZ3
VPC	vpc-kafka
Subnet	subnet-kafka
Security Group	sg-kafka

Step 7 Click the VPC to obtain the VPC ID on the VPC console.

Figure 4-20 Obtaining the VPC ID



VPC Information

Name	vpc-kafka
ID	7066[redacted]8f2c888
Status	Available
CIDR Block	192.168.0.0/16 Edit CIDR Block

Step 8 Call the VPC Endpoint API to create a VPC endpoint service. For details, see **Creating a VPC Endpoint Service**.

```
curl -i -k -H 'Accept:application/json' -H 'Content-Type:application/json;charset=utf8' -X POST -H "X-Auth-Token:$token" -d '{"port_id":"38axxxeac","vpc_id":"706xxx888","ports":[{"protocol":"TCP","client_port":
```

```
9011,"server_port":9011 }],"approval_enabled":false,"service_type":"interface","server_type":"VM"}' https://
{endpoint}/v1/{project_id}/vpc-endpoint-services
```

Parameter description:

- **token**: an access credential issued to an IAM user to bear its identity and permissions. For details on how to obtain a token, see **Obtaining a User Token**.
- **port_id**: one of the port IDs obtained in **Step 5**.
- **vpc_id**: VPC ID obtained in **Step 7**.
- **endpoint**: VPCEP endpoint obtained from Regions and Endpoints. The region must be the same as that of the Kafka instance.
- **project_id**: project ID obtained from **Obtaining a Project ID**. The region must be the same as that of the Kafka instance.

Record the value of **service_name** in the response. This parameter indicates the name of the VPC endpoint service.

Step 9 Repeat **Step 8** to create VPC endpoint services for other port IDs obtained in **Step 5** and record the VPC endpoint service names.

----End

(Optional) Adding a Whitelist Record

If the Kafka client and Kafka instance belong to different accounts, add the ID of the account to which the Kafka client belongs to the whitelist of the endpoint service. For details, see **Add a Whitelist Record**.

Buying a VPC Endpoint

Step 1 Click ☰ in the upper left corner of the management console. Then choose **Network** > **VPC Endpoint**.

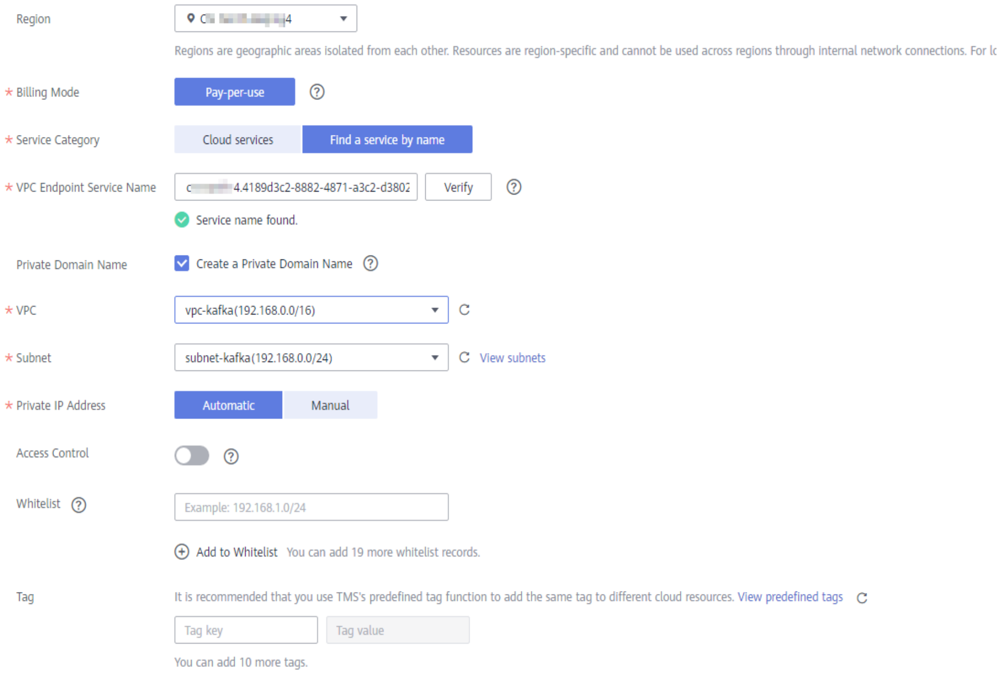
Step 2 Click **Buy VPC Endpoint**.

Step 3 Set the following parameters:

- **Region**: Select the region that the Kafka instance is in.
- **Service Category**: Select **Find a service by name**.
- **VPC Endpoint Service Name**: Enter the VPC endpoint service name recorded in **Step 8** and click **Verify**. If **Service name found** is displayed, proceed with subsequent operations.
- **VPC**: Select the VPC that the Kafka client is in.
- **Subnet**: Select the subnet that the Kafka client is in.
- **Private IP Address**: Select **Automatic**.

Retain the default values for other parameters. For details, see **Buying a VPC Endpoint**.

Figure 4-21 VPC endpoint parameters



Region: C[redacted]4

Regions are geographic areas isolated from each other. Resources are region-specific and cannot be used across regions through internal network connections. For lc

* Billing Mode: Pay-per-use

* Service Category: Cloud services | Find a service by name

* VPC Endpoint Service Name: c[redacted]4.4189d3c2-8882-4871-a3c2-d3802 Verify

Service name found.

Private Domain Name: ☑ Create a Private Domain Name

* VPC: vpc-kafka(192.168.0.0/16)

* Subnet: subnet-kafka(192.168.0.0/24) View subnets

* Private IP Address: Automatic | Manual

Access Control

Whitelist: Example: 192.168.1.0/24

Add to Whitelist You can add 19 more whitelist records.

Tag: It is recommended that you use TMS's predefined tag function to add the same tag to different cloud resources. View predefined tags

Tag key | Tag value

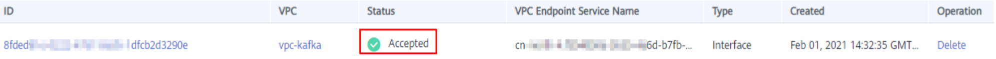
You can add 10 more tags.

Step 4 Click **Next**.

Step 5 Confirm the configurations and submit the request.

Step 6 Go back to the VPC endpoint list and check whether the status of the created VPC endpoint has changed to **Accepted**. The **Accepted** state means that the VPC endpoint has been connected to the VPC endpoint service.

Figure 4-22 Checking the VPC endpoint status

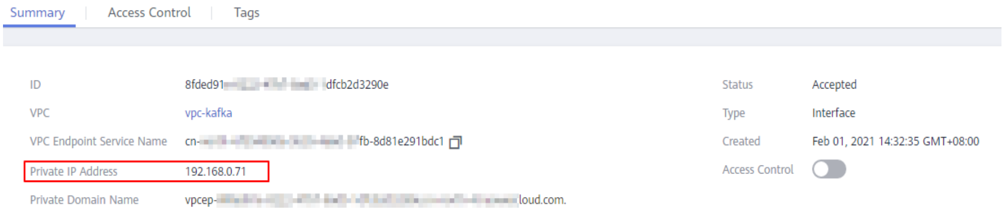


ID	VPC	Status	VPC Endpoint Service Name	Type	Created	Operation
8fded[redacted]dfcb2d3290e	vpc-kafka	Accepted	cn[redacted]6d-b7fb-...	Interface	Feb 01, 2021 14:32:35 GMT...	Delete

Step 7 Click the VPC endpoint ID. On the **Summary** tab page, obtain the private IP address.

You can use the private IP address to access the VPC endpoint service.

Figure 4-23 Viewing the private IP address



Summary | Access Control | Tags

ID	8fded91[redacted]dfcb2d3290e	Status	Accepted
VPC	vpc-kafka	Type	Interface
VPC Endpoint Service Name	cn-[redacted]fb-8d81e291bdc1	Created	Feb 01, 2021 14:32:35 GMT+08:00
Private IP Address	192.168.0.71	Access Control	
Private Domain Name	vpcep-[redacted]loud.com.		

Step 8 Repeat **Step 1** to **Step 7** to buy a VPC endpoint for each VPC endpoint service created in **Step 9**, and view and record the private IP addresses of the VPC endpoint services.

----End

Changing the advertised.listeners IP Address

Step 1 Click ☰ and choose **Application** > **Distributed Message Service (for Kafka)** to open the console of DMS for Kafka.

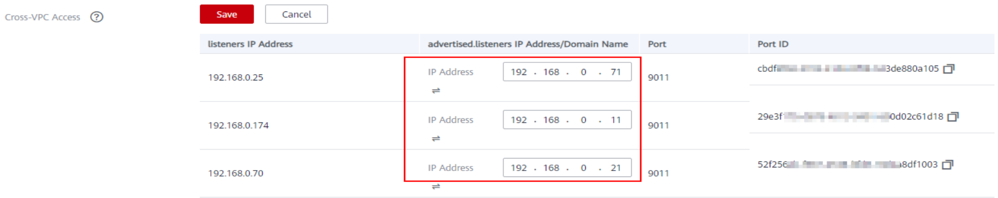
Step 2 Click the desired Kafka instance to view the instance details.

Step 3 On the **Advanced Settings** section of the **Basic Information** tab page, click **Modify** for **Cross-VPC Access** to change the value of **advertised.listeners IP address** to the private IP addresses recorded in **Step 7** and **Step 8**. Click **Save**.

NOTICE

Each IP address must match the corresponding port ID. Otherwise, the network will be disconnected.

Figure 4-24 Changing the advertised.listeners IP addresses



Cross-VPC Access ?

Save Cancel

listeners IP Address	advertised.listeners IP Address/Domain Name	Port	Port ID
192.168.0.25	IP Address 192 . 168 . 0 . 71 ⇌	9011	cbdf[redacted]3de880a105
192.168.0.174	IP Address 192 . 168 . 0 . 11 ⇌	9011	29e3f[redacted]0d02c61d18
192.168.0.70	IP Address 192 . 168 . 0 . 21 ⇌	9011	52f256[redacted]8df1003

----End

Verifying Connectivity

Check whether messages can be created and retrieved by referring to **Accessing a Kafka Instance Without SASL** or **Accessing a Kafka Instance with SASL**.

Notes:

- The address for connecting to a Kafka instance is in the format of "*advertised.listeners IP*:**9011**". For example, the addresses for connecting to the Kafka instance shown in **Figure 4-24** are **192.168.0.71:9011,192.168.0.11:9011,192.168.0.21:9011**.
- Configure inbound rules for the security group of the Kafka instance to allow access from **198.19.128.0/17** over port **9011**.
- If a network access control list (ACL) has been configured for the subnet of this instance, configure inbound rules for the network ACL to allow access from **198.19.128.0/17** and from the subnet used by the VPC endpoint.

NOTE

198.19.128.0/17 is the network segment allocated to the VPCEP service. To use VPCEP, allow access from this network segment.

4.5 Using DNAT to Access a Kafka Instance

Scenario

You can use destination NAT (DNAT) to access a Kafka instance so that the instance can provide services on the public network through port mapping.

Prerequisites

You have purchased EIPs. The number of EIPs is the same as the number of brokers in the Kafka instance.

Step 1: Obtain Information About the Kafka Instance

Step 1 Log in to the management console.

Step 2 Click 📍 in the upper left corner to select a region.

NOTE

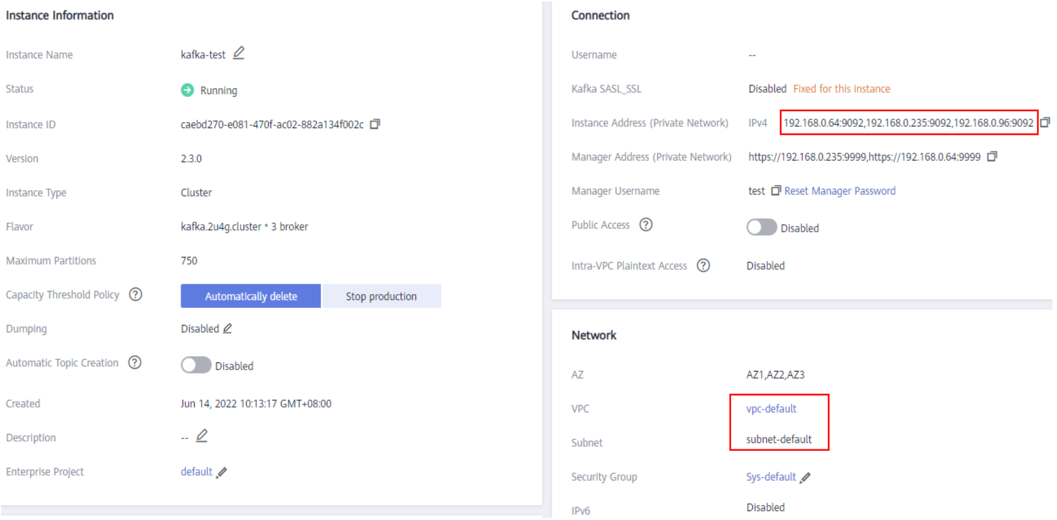
Select the region where your Kafka instance is located.

Step 3 Click ☰ and choose **Application** > **Distributed Message Service (for Kafka)** to open the console of DMS for Kafka.

Step 4 Click the desired Kafka instance to view the instance details.

Step 5 In the **Connection** area on the **Basic Information** tab page, view and record the private network access addresses of the Kafka instance. In the **Network** area, view and record the VPC and subnet where the Kafka instance is located.

Figure 4-25 Kafka instance information



Instance Information

Instance Name	kafka-test
Status	Running
Instance ID	caebd270-e081-470f-ac02-882a134f002c
Version	2.3.0
Instance Type	Cluster
Flavor	kafka.2u4g.cluster * 3 broker
Maximum Partitions	750
Capacity Threshold Policy	Automatically delete / Stop production
Dumping	Disabled
Automatic Topic Creation	Disabled
Created	Jun 14, 2022 10:13:17 GMT+08:00
Description	--
Enterprise Project	default

Connection

Username	--
Kafka SASL_SSL	Disabled Fixed for this instance
Instance Address (Private Network)	IPv4 192.168.0.64:9092,192.168.0.235:9092,192.168.0.96:9092
Manager Address (Private Network)	https://192.168.0.235:9999,https://192.168.0.64:9999
Manager Username	test Reset Manager Password
Public Access	Disabled
Intra-VPC Plaintext Access	Disabled

Network

AZ	AZ1,AZ2,AZ3
VPC	vpc-default
Subnet	subnet-default
Security Group	Sys-default
IPv6	Disabled

----End

Step 2: Buy a Public NAT Gateway

Step 1 Click ☰ in the upper left corner of the management console and choose **Network** > **NAT Gateway**. The **Public NAT Gateways** page is displayed.

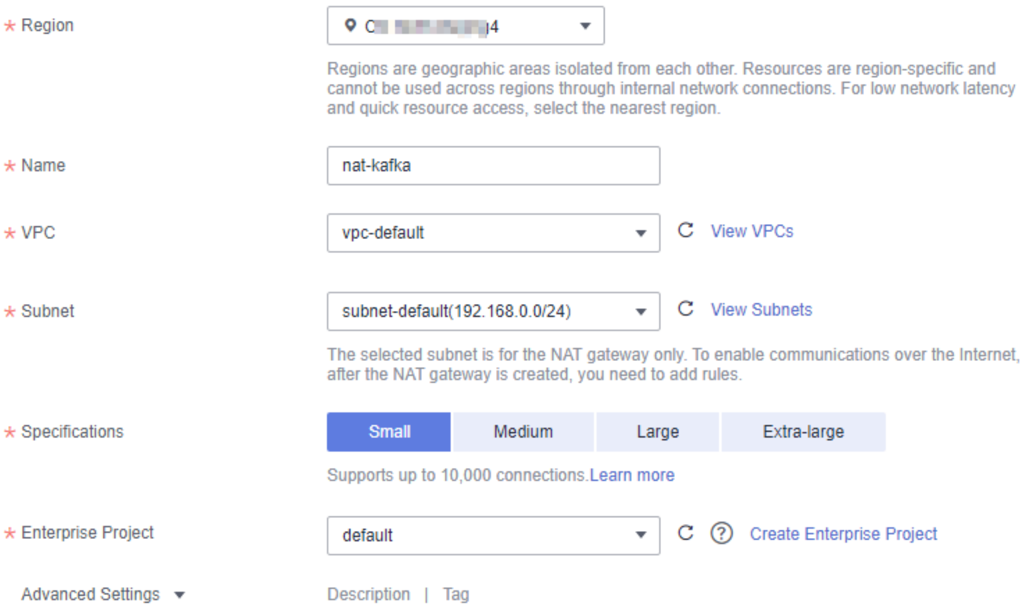
Step 2 Click **Buy Public NAT Gateway**.

Step 3 Set the following parameters:

- **Region**: Select the region that the Kafka instance is in.
- **Name**: Enter a name for the public NAT gateway.
- **VPC**: Select the VPC recorded in **Step 5**.
- **Subnet**: Select the subnet recorded in **Step 5**.
- **Enterprise Project**: Select an enterprise project as required.

Set other parameters as required. For details, see **Creating a Public NAT Gateway**.

Figure 4-26 Buying a public NAT gateway



* Region: C[redacted]4

Regions are geographic areas isolated from each other. Resources are region-specific and cannot be used across regions through internal network connections. For low network latency and quick resource access, select the nearest region.

* Name: nat-kafka

* VPC: vpc-default

* Subnet: subnet-default(192.168.0.0/24)

The selected subnet is for the NAT gateway only. To enable communications over the Internet, after the NAT gateway is created, you need to add rules.

* Specifications: Small

Supports up to 10,000 connections.Learn more

* Enterprise Project: default

Step 4 Click **Next**.

Step 5 Confirm the specifications and click **Submit**.

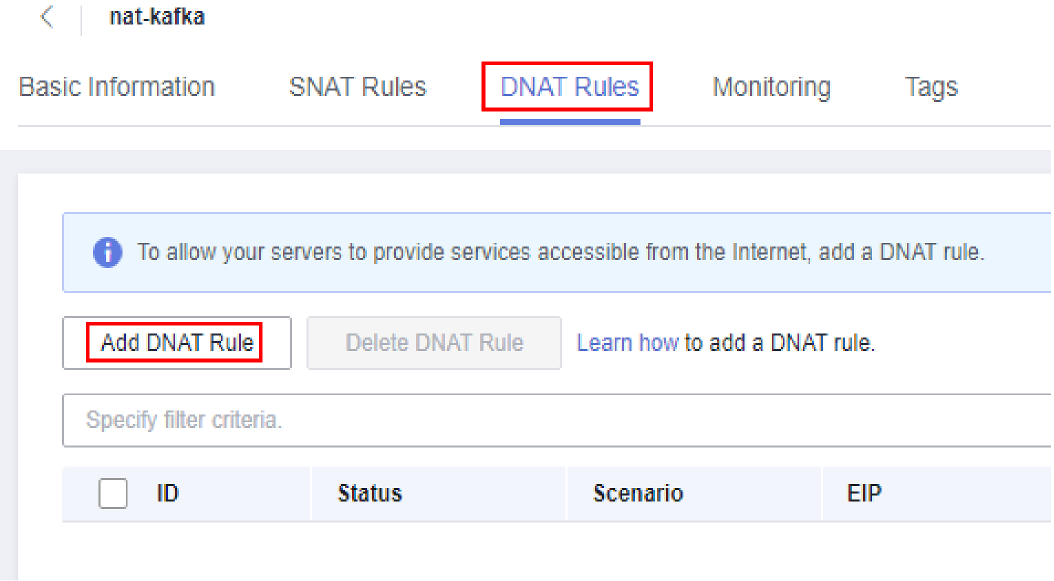
----End

Step 3: Add a DNAT Rule

Step 1 On **Public NAT Gateways** page, locate the row that contains the newly purchased public NAT gateway and click **Configure Rules** in the **Operation** column.

Step 2 On the **DNAT Rules** tab page, click **Add DNAT Rule**.

Figure 4-27 Public NAT gateway details



To allow your servers to provide services accessible from the Internet, add a DNAT rule.

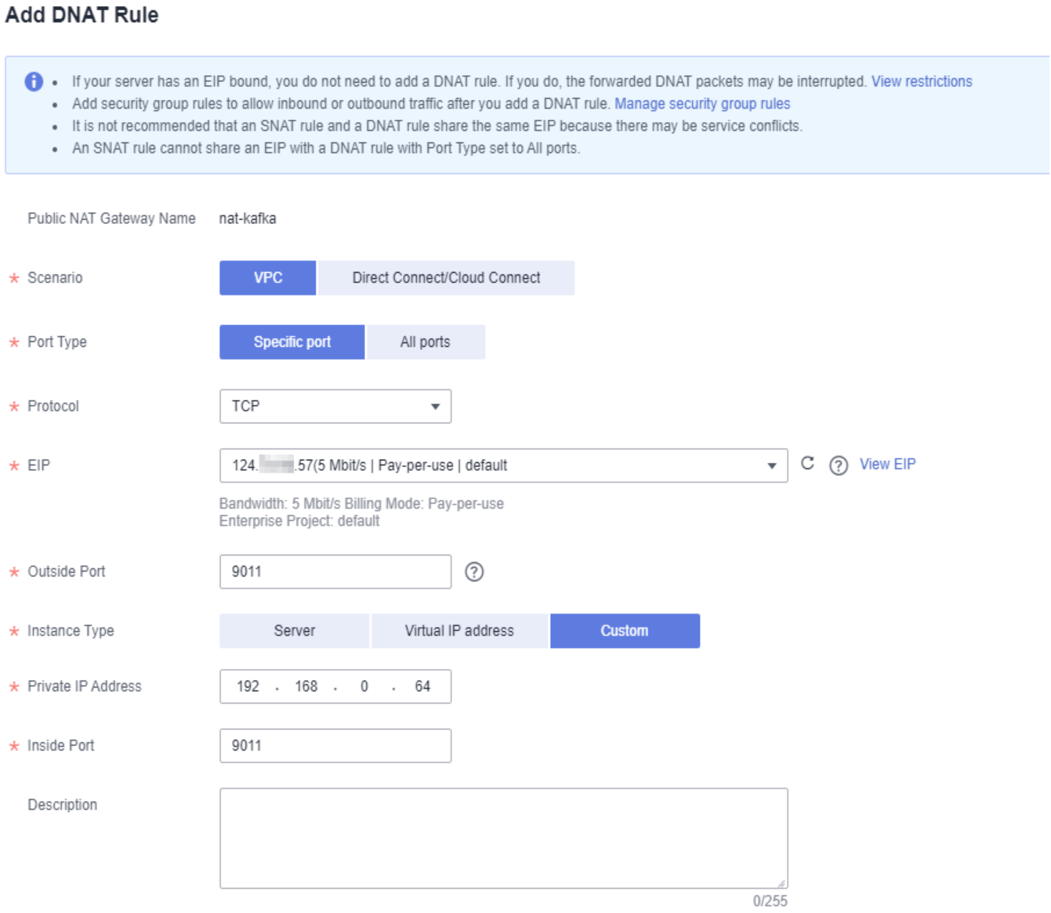
Learn how to add a DNAT rule.

Step 3 Set the following parameters:

- **Scenario**: Select **VPC**.
- **Port Type**: Select **Specific port**.
- **Protocol**: Select **TCP**.
- **EIP**: Select an EIP.
- **Outside Port**: Enter **9011**.
- **Instance Type**: Select **Custom**.
- **Private IP Address**: Enter one of the private network addresses of the Kafka instance recorded in **Step 5**.
- **Inside Port**: Enter **9011**.

For details about more parameters, see **Adding a DNAT Rule**.

Figure 4-28 Adding a DNAT rule



Add DNAT Rule

- If your server has an EIP bound, you do not need to add a DNAT rule. If you do, the forwarded DNAT packets may be interrupted. View restrictions
- Add security group rules to allow inbound or outbound traffic after you add a DNAT rule. Manage security group rules
- It is not recommended that an SNAT rule and a DNAT rule share the same EIP because there may be service conflicts.
- An SNAT rule cannot share an EIP with a DNAT rule with Port Type set to All ports.

Public NAT Gateway Name: nat-kafka

* Scenario: VPC | Direct Connect/Cloud Connect

* Port Type: Specific port | All ports

* Protocol: TCP

* EIP: 124.[redacted].57(5 Mbit/s | Pay-per-use | default

Bandwidth: 5 Mbit/s Billing Mode: Pay-per-use
Enterprise Project: default

* Outside Port: 9011

* Instance Type: Server | Virtual IP address | Custom

* Private IP Address: 192 . 168 . 0 . 64

* Inside Port: 9011

Description:

0/255

Step 4 Click **OK**.

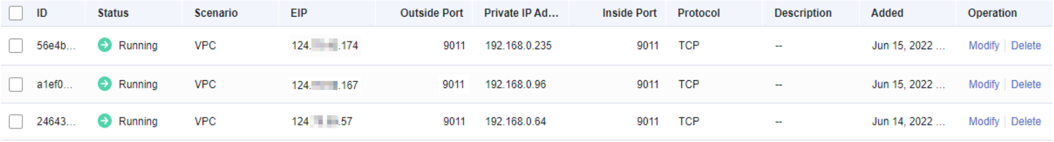
View the DNAT rule status in the DNAT rule list. If **Status** is **Running**, the rule has been added successfully.

Step 5 Create DNAT rules for other private network addresses of the Kafka instance recorded in Step 5. **Configure a unique EIP for each DNAT rule.**

For details about how to create a DNAT rule, see Step 2 to Step 4.

Step 6 After all DNAT rules are created, click the **DNAT Rules** tab to view the created DNAT rules and record the EIPs corresponding to the private IP addresses.

Figure 4-29 DNAT rule list



□	ID	Status	Scenario	EIP	Outside Port	Private IP Ad...	Inside Port	Protocol	Description	Added	Operation
□	56e4b...	Running	VPC	124.[redacted].174	9011	192.168.0.235	9011	TCP	--	Jun 15, 2022 ...	Modify | Delete
□	a1ef0...	Running	VPC	124.[redacted].167	9011	192.168.0.96	9011	TCP	--	Jun 15, 2022 ...	Modify | Delete
□	24643...	Running	VPC	124[redacted].57	9011	192.168.0.64	9011	TCP	--	Jun 14, 2022 ...	Modify | Delete

----End

Step 4: Bind EIPs on the Kafka Console

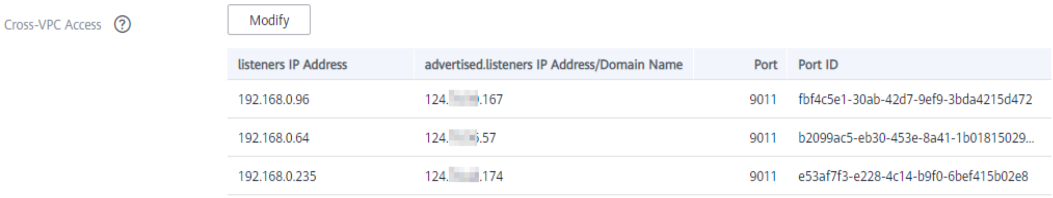
Step 1 Click ☰ and choose **Application** > **Distributed Message Service (for Kafka)** to open the console of DMS for Kafka.

Step 2 Click the desired Kafka instance to view the instance details.

Step 3 In the **Advanced Settings** section on the **Basic Information** tab page, click **Modify** next to **Cross-VPC Access**.

Step 4 Change the values of **advertised.listeners IP Address/Domain Name** to the EIPs in the DNAT rules. Ensure that the mapping between the private network addresses and the EIPs is consistent with that recorded in **Step 6**. Then click **Save**.

Figure 4-30 Changing the advertised.listeners IP address (for DNAT access)



listeners IP Address	advertised.listeners IP Address/Domain Name	Port	Port ID
192.168.0.96	124.[redacted].167	9011	fbf4c5e1-30ab-42d7-9ef9-3bda4215d472
192.168.0.64	124.[redacted].57	9011	b2099ac5-eb30-453e-8a41-1b01815029...
192.168.0.235	124.[redacted].174	9011	e53af7f3-e228-4c14-b9f0-6bef415b02e8

----End

Step 5: Verify Connectivity

Check whether messages can be created and retrieved by referring to **Accessing a Kafka Instance Without SASL** or **Accessing a Kafka Instance with SASL**.

Notes:

- The address for connecting to a Kafka instance is in the format of "*advertised.listeners IP*:**9011**". For example, the addresses for connecting to the Kafka instance shown in **Figure 4-30** are **124.*xxx.xxx*.167:9011,124.*xxx.xxx*.174:9011,124.*xxx.xxx*.57:9011**.
- Configure security group rules for the Kafka instance to allow inbound access over port **9011**.
- Public access must be enabled on the client connected to the Kafka instance.

5 Managing Instances

5.1 Modifying Instance Specifications

Scenario

After creating a Kafka instance, you can expand its storage space and increase the number of brokers.

Notes and Constraints

- You can expand the storage space 20 times.
- If you increase the number of brokers, the maximum number of partitions will also be increased. When brokers are added, the storage space is proportionally expanded based on the current disk space. For example, assume that the original number of brokers of an instance is 3 and the disk size of each broker is 200 GB. If the broker quantity changes to 10 and the disk size of each broker is still 200 GB, the total disk size becomes 2000 GB.

Procedure

Step 1 Log in to the management console.

Step 2 Click 📍 in the upper left corner to select a region.

NOTE

Select the region where your Kafka instance is located.

Step 3 Click ☰ and choose **Application** > **Distributed Message Service (for Kafka)** to open the console of DMS for Kafka.

Step 4 In the row containing the instance for which you want to modify the specifications, choose **More** > **Modify Specifications** in the **Operation** column.

Step 5 Specify the required storage space or the number of brokers.

The storage space and the number of brokers can only be changed separately.

- Expand the storage space.

  For **Modify By**, select **Storage**. For **Storage Space per Broker**, specify a new storage space, and click **Next**. Confirm the configurations and click **Submit**.

  View the new storage space (Storage space per broker x Number of brokers) in the **Used/Available Storage Space (GB)** column in the instance list.

  NOTE

  – Storage space expansion does not affect services.

  – Available storage space = Actual storage space – Storage space for storing logs and ZooKeeper data – Disk formatting loss

    For example, if the storage space is expanded to 700 GB, the storage space for storing logs and ZooKeeper data is 100 GB, and the disk formatting loss is 7 GB, then the available storage space after capacity expansion will be 593 GB.

- Add brokers.

  For **Modify By**, select **Brokers**. Then, enter the number of brokers and click **Next**. Confirm the configurations and click **Submit**.

  View the number of brokers in the **Specifications** column in the instance list.

  NOTE

  – Adding brokers does not affect the original brokers or services.

  – New topics are created on new brokers, and the original topics are still on the original brokers, resulting in unbalanced partitions. You can **reassign partitions** to migrate the replicas of the original topic partitions to the new brokers.

  – After adding brokers, add the IP addresses of the new brokers to the client connection configuration to improve reliability.

----End

5.2 Viewing an Instance

Scenario

View detailed information about a Kafka instance on the Kafka console, for example, the IP addresses and port numbers for accessing the instance.

Procedure

Step 1 Log in to the management console.

Step 2 Click 📍 in the upper left corner to select a region.

NOTE

Select the region where your Kafka instance is located.

Step 3 Click ☰ and choose **Application** > **Distributed Message Service (for Kafka)** to open the console of DMS for Kafka.

Step 4 Search for a Kafka instance by tag, status, name, ID, or connection address. **Table 5-1** describes the various possible statuses of a Kafka instance.

Table 5-1 Kafka instance status description

Status	Description
Creating	The instance is being created.
Running	The instance is running properly. Only instances in the **Running** state can provide services.
Faulty	The instance is not running properly.
Starting	The status between **Frozen** and **Running**.
Restarting	The instance is being restarted.
Changing	The instance specifications or public access configurations are being modified.
Change failed	The instance specifications or public access configurations failed to be modified.
Frozen	The instance is frozen.
Freezing	The status between **Running** and **Frozen**.
Upgrading	The instance is being upgraded.
Rolling back	The instance is being rolled back.

Step 5 Click the name of the desired Kafka instance and view detailed information about the instance on the **Basic Information** tab page.

Table 5-2 describes the parameters for connecting to a Kafka instance. For details about other parameters, see the **Basic Information** tab page of the Kafka instance on the console.

Table 5-2 Connection parameters

Section	Parameter	Description
Connection	Username	Username for accessing the instance with SASL_SSL enabled.
	Kafka SASL_SSL	Whether SASL_SSL is enabled.
	SASL Mechanism	SASL mechanism used by the instance with SASL_SSL enabled.
	SSL Certificate	Click **Download** to download the SSL certificate for accessing the instance.

Section	Parameter	Description
	Instance Address (Private Network)	Address for connecting to the instance when public access is disabled. The number of connection addresses is the same as that of brokers.
	Manager Address (Private Network)	Address for connecting to Kafka Manager when public access is disabled.
	Manager Username	Username for connecting to Kafka Manager.
	Public Access	Indicates whether public access has been enabled for the instance.
	Instance Address (Public Network)	Address for connecting to the instance when public access is enabled. This parameter is displayed only when public access is enabled.
	Manager Address (Public Network)	Address for connecting to Kafka Manager when public access is enabled. This parameter is displayed only when public access is enabled.
	Intra-VPC Plaintext Access	Whether intra-VPC plaintext access is enabled.

----End

5.3 Restarting an Instance

Scenario

Restart one or more Kafka instances at a time on the Kafka console.

NOTICE

When a Kafka instance is being restarted, message retrieval and creation requests of clients will be rejected.

Prerequisites

The status of the Kafka instance you want to restart is either **Running** or **Faulty**.

Procedure

Step 1 Log in to the management console.

Step 2 Click 📍 in the upper left corner to select a region.

NOTE

Select the region where your Kafka instance is located.

Step 3 Click ☰ and choose **Application** > **Distributed Message Service (for Kafka)** to open the console of DMS for Kafka.

Step 4 Restart Kafka instances using one of the following methods:

- Select one or more Kafka instances and click **Restart** in the upper left corner.
- In the row containing the desired instance, click **Restart**.
- Click the desired Kafka instance to view the instance details. In the upper right corner, click **Restart**.

Step 5 In the **Restart Instance** dialog box, click **Yes** to restart the Kafka instance.

It takes 3 to 15 minutes to restart a Kafka instance. After the instance is successfully restarted, its status should be **Running**.

NOTE

Restarting a Kafka instance only restarts the instance process and does not restart the VM where the instance is located.

----End

5.4 Deleting an Instance

Scenario

On the Kafka console, you can delete one or more Kafka instances that have been created or failed to be created.

NOTICE

Deleting a Kafka instance will delete the data in the instance without any backup. Exercise caution when performing this operation.

Prerequisites

The status of the Kafka instance you want to delete is **Running** or **Faulty**.

Deleting Kafka Instances

Step 1 Log in to the management console.

Step 2 Click 📍 in the upper left corner to select a region.

NOTE

Select the region where your Kafka instance is located.

Step 3 Click ☰ and choose **Application** > **Distributed Message Service (for Kafka)** to open the console of DMS for Kafka.

Step 4 Delete Kafka instances using one of the following methods:

- Select one or more Kafka instances and click **Delete** in the upper left corner.
- In the row containing the Kafka instance to be deleted, choose **More** > **Delete**.
- Click the desired Kafka instance to view its details. In the upper right corner, choose **More** > **Delete**.

NOTE

Kafka instances in the **Creating**, **Starting**, **Changing**, **Change failed**, or **Restarting** state cannot be deleted.

Step 5 In the **Delete Instance** dialog box, click **Yes** to delete the Kafka instance.

It takes 1 to 60 seconds to delete a Kafka instance.

----End

Deleting Kafka Instances That Failed to Be Created

Step 1 Log in to the management console.

Step 2 Click 📍 in the upper left corner to select a region.

NOTE

Select the region where your Kafka instance is located.

Step 3 Click ☰ and choose **Application** > **Distributed Message Service (for Kafka)** to open the console of DMS for Kafka.

Step 4 If there are Kafka instances that failed to be created, **Instance Creation Failures** and quantity information will be displayed.

NOTE

Instances that fail to be created do not occupy other resources.

Step 5 Click **Instance Creation Failures** or the icon or quantity next to it.

Step 6 Delete Kafka instances that failed to be created in either of the following ways:

- To delete all Kafka instances that failed to be created at once, click **Clear Failed Instance**.
- To delete a single Kafka instance that failed to be created, click **Delete** in the row containing the chosen Kafka instance.

----End

5.5 Modifying the Information About an Instance

After creating a Kafka instance, you can modify some parameters of the instance based on service requirements, including the instance name, description, security group, and capacity threshold policy.

Procedure

Step 1 Log in to the management console.

Step 2 Click 📍 in the upper left corner to select a region.

> **NOTE**
>
> Select the region where your Kafka instance is located.

Step 3 Click ☰ and choose **Application** > **Distributed Message Service (for Kafka)** to open the console of DMS for Kafka.

Step 4 Click the desired Kafka instance to view the instance details.

Step 5 Modify the following parameters if needed:

- Instance Name
- Enterprise Project (Changing the enterprise project will not restart the instance.)
- Description
- Security Group
- Public Network Access (For details about how to change the public network access configuration, see **Configuring Public Access**.)
- Capacity Threshold Policy (Modifying this setting will not restart the instance.)
- Automatic Topic Creation (Modifying this setting will restart the instance.)
- Cross-VPC Access (See **Cross-VPC Access to a Kafka Instance** and **Using DNAT to Access a Kafka Instance**.)

After the parameters are modified, view the modification result in one of the following ways:

- If **Capacity Threshold Policy**, **Public Network Access**, or **Automatic Topic Creation** has been modified, you will be redirected to the **Background Tasks** page, which displays the modification progress and result.
- If **Instance Name**, **Description**, **Enterprise Project**, **Cross-VPC Access**, or **Security Group** has been modified, the modification result will be displayed in the upper right corner of the page.

----End

5.6 Configuring Public Access

To access a Kafka instance over a public network, enable public access and configure EIPs for the instance.

If you no longer need public access to the instance, you can disable it as required.

Prerequisites

- You can change the public access setting only when the Kafka instance is in the **Running** state.
- Only IPv4 EIPs can be bound to Kafka instances.

Enabling Public Network Access

Step 1 Log in to the management console.

Step 2 Click 📍 in the upper left corner to select a region.

NOTE

Select the region where your Kafka instance is located.

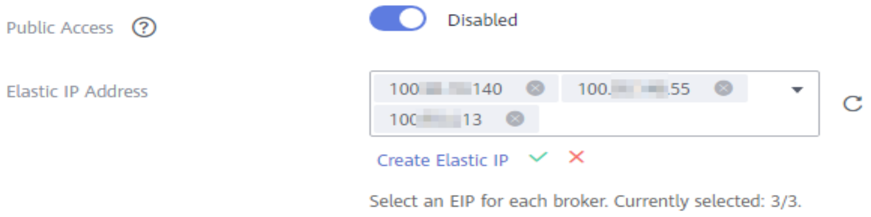
Step 3 Click ☰ and choose **Application** > **Distributed Message Service (for Kafka)** to open the console of DMS for Kafka.

Step 4 Click a Kafka instance to go to the **Basic Information** tab page.

Step 5 Click ○ next to **Public Access** to enable public access. For **Elastic IP Address**, select an EIP for each broker and then click ✓.

You can view the operation progress on the **Background Tasks** page. If the task status is **Successful**, the modification has succeeded.

Figure 5-1 Enabling public access



Public Access ?
Disabled

Elastic IP Address
100[redacted]140 ⊗ 100.[redacted]55 ⊗
100[redacted]13 ⊗

Create Elastic IP ✓ ✕

Select an EIP for each broker. Currently selected: 3/3.

After public access is enabled, configure security group rules listed in **Table 5-3** before attempting to access Kafka. For details about accessing Kafka, see **Accessing a Kafka Instance**.

Table 5-3 Security group rules (public network access)

Direction	Protocol	Port	Source	Description
Inbound	TCP	9094	0.0.0.0/0	Access Kafka through the public network (without SSL encryption).

Direction	Protocol	Port	Source	Description
Inbound	TCP	9095	0.0.0.0/0	Access Kafka through the public network (with SSL encryption).

----End

Disabling Public Network Access

Step 1 Log in to the management console.

Step 2 Click 📍 in the upper left corner to select a region.

NOTE

Select the region where your Kafka instance is located.

Step 3 Click ☰ and choose **Application** > **Distributed Message Service (for Kafka)** to open the console of DMS for Kafka.

Step 4 Click a Kafka instance to go to the **Basic Information** tab page.

Step 5 Click 🔘 next to **Public Access**.

You can view the operation progress on the **Background Tasks** page. If the task status is **Successful**, the modification has succeeded.

After public access is disabled, configure security group rules listed in **Table 5-4** before attempting to access Kafka in a VPC. For details about accessing Kafka, see **Accessing a Kafka Instance**.

Table 5-4 Security group rules (private network access)

Direction	Protocol	Port	Source	Description
Inbound	TCP	9092	0.0.0.0/0	Access a Kafka instance within a VPC (without SSL encryption).
Inbound	TCP	9093	0.0.0.0/0	Access a Kafka instance within a VPC (with SSL encryption).

NOTE

After a security group is created, its default inbound rule allows communication among ECSs within the security group and its default outbound rule allows all outbound traffic. In this case, you can access a Kafka instance within a VPC, and do not need to add rules according to **Table 5-4**.

----End

5.7 Resetting Kafka Password

Scenario

For a Kafka instance with SASL_SSL enabled, there are two ways to create an SASL_SSL user on the console. Accordingly, there are two ways to reset the SASL_SSL user's password:

- If an SASL_SSL user is created during instance creation, reset their password by referring to the following instructions.
- If an SASL_SSL user is created on the **Users** tab page, reset their password by referring to **Resetting the SASL_SSL Password**.

Prerequisites

- You can reset the Kafka password only if Kafka SASL_SSL has been enabled for the instance.
- You can reset the Kafka password only when the instance is in the **Running** state.

Procedure

Step 1 Log in to the management console.

Step 2 Click 📍 in the upper left corner to select a region.

NOTE

Select the region where your Kafka instance is located.

Step 3 Click ☰ and choose **Application** > **Distributed Message Service (for Kafka)** to open the console of DMS for Kafka.

Step 4 Reset the Kafka instance password using either of the following methods:

- Choose **More** > **Reset Kafka Password** in the row containing the desired Kafka instance.
- Click the desired Kafka instance to view its details. On the **Basic Information** tab page, click **Reset Password** next to **Username** in the **Connection** section.
- Click the desired Kafka instance to view its details. On the **Users** tab page, click **Reset Password** in the row containing the desired user.

Step 5 In the **Reset Kafka Password** dialog box, enter and confirm a new password, and click **OK**.

- If the password is successfully reset, a success message is displayed.
- If the password fails to be reset, a failure message is displayed. Reset the password again. If you still fail to reset the password after multiple attempts, contact customer service.

NOTE

The system will display a success message only after the password is successfully reset on all brokers.

----End

5.8 Resetting Kafka Manager Password

Scenario

You can reset the password of Kafka Manager of a Kafka instance if you forget it.

Prerequisites

A Kafka instance has been created and is in the **Running** state.

Procedure

Step 1 Log in to the management console.

Step 2 Click 📍 in the upper left corner to select a region.

NOTE

Select the region where your Kafka instance is located.

Step 3 Click ☰ and choose **Application** > **Distributed Message Service (for Kafka)** to open the console of DMS for Kafka.

Step 4 Reset the Kafka Manager password using either of the following methods:

- In the row containing the desired Kafka instance, choose **More** > **Reset Manager Password**.
- Click the desired Kafka instance to view its details. In the upper right corner, choose **More** > **Reset Manager Password**.
- Click the desired Kafka instance to view its details. On the **Basic Information** tab page, click **Reset Manager Password** next to **Manager Username** in the **Connection** section.

Step 5 Enter and confirm a new password, and click **OK**.

- If the password is successfully reset, a success message is displayed.
- If the password fails to be reset, a failure message is displayed. Reset the password again. If you still fail to reset the password after multiple attempts, contact customer service.

NOTE

The system will display a success message only after the password is successfully reset on all brokers.

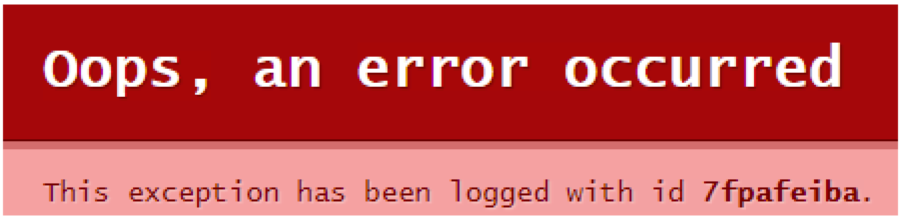
----End

5.9 Restarting Kafka Manager

Scenario

Restart Kafka Manager when you fail to log in to it or it cannot provide services as usual.

Figure 5-2 Error information



Oops, an error occurred

This exception has been logged with id **7fpafeiba**.

NOTE

Restarting Kafka Manager does not affect services.

Procedure

Step 1 Log in to the management console.

Step 2 Click 📍 in the upper left corner to select a region.

NOTE

Select the region where your Kafka instance is located.

Step 3 Click ☰ and choose **Application** > **Distributed Message Service (for Kafka)** to open the console of DMS for Kafka.

Step 4 Restart Kafka Manager using either of the following methods:

- In the row containing the desired Kafka instance, choose **More** > **Restart Kafka Manager**.
- Click the desired Kafka instance to view the instance details. In the upper right corner, choose **More** > **Restart Kafka Manager**.

Step 5 Click **Yes**.

You can view the operation progress on the **Background Tasks** page. If the task status is **Successful**, the restart has succeeded.

----End

5.10 Managing Instance Tags

Tags facilitate Kafka instance identification and management.

You can add tags to a Kafka instance when creating the instance or add tags on the **Tags** tab page of the created instance. Up to 20 tags can be added to an instance. Tags can be modified and deleted.

A tag consists of a tag key and a tag value. **Table 5-5** lists the tag key and value requirements.

Table 5-5 Tag key and value requirements

Parameter	Requirements
Tag key	● Cannot be left blank. ● Must be unique for the same instance. ● Can contain a maximum of 36 characters. ● Cannot contain the following characters: =*<>\,|/ ● Cannot start or end with a space.
Tag value	● Cannot be left blank. ● Can contain a maximum of 43 characters. ● Cannot contain the following characters: =*<>\,|/ ● Cannot start or end with a space.

Procedure

Step 1 Log in to the management console.

Step 2 Click 📍 in the upper left corner to select a region.

NOTE

Select the region where your Kafka instance is located.

Step 3 Click ☰ and choose **Application** > **Distributed Message Service (for Kafka)** to open the console of DMS for Kafka.

Step 4 Click the name of an instance.

Step 5 Click the **Tags** tab.

View the tags of the instance.

Step 6 Perform the following operations as required:

- Add a tag
  1. Click **Create/Delete Tag**.
  2. Enter a tag key and a tag value, and click **Add**.

     If you have predefined tags, select a predefined pair of tag key and value, and click **Add**.
  3. Click **OK**.
- Delete a tag

  Delete a tag using either of the following methods:
  - In the row containing the tag to be deleted, click **Delete**. In the **Delete Tag** dialog box, click **Yes**.
  - Click **Create/Delete Tag**. In the dialog box that is displayed, click ⊗ next to the tag to be deleted and click **OK**.

----End

5.11 Viewing Background Tasks

After you initiate certain instance operations such as configuring public access and modifying the capacity threshold policy, a background task will start for each operation. On the console, you can view the background task status and clear task information by deleting task records.

Procedure

Step 1 Log in to the management console.

Step 2 Click 📍 in the upper left corner to select a region.

NOTE

Select the region where your Kafka instance is located.

Step 3 Click ☰ and choose **Application** > **Distributed Message Service (for Kafka)** to open the console of DMS for Kafka.

Step 4 Click a Kafka instance to go to the **Basic Information** tab page.

Step 5 Click the **Background Tasks** tab.

Step 6 In the upper right corner, click the time period next to the calendar icon, select the start time and end time, and click **OK**. Tasks started in the specified period are displayed.

On the **Background Tasks** page, you can also perform the following operations:

- Click ⟳ to refresh the task status.
- Click **Delete**. In the displayed **Delete Task** dialog box, click **Yes** to clear the task information.

NOTE

You can only delete the records of tasks in the **Successful** or **Failed** state.

----End

5.12 Viewing Disk Usage

On the Kafka console, you can view the disk usage of each broker.

Procedure

Step 1 Log in to the management console.

Step 2 Click in the upper left corner to select a region.

NOTE

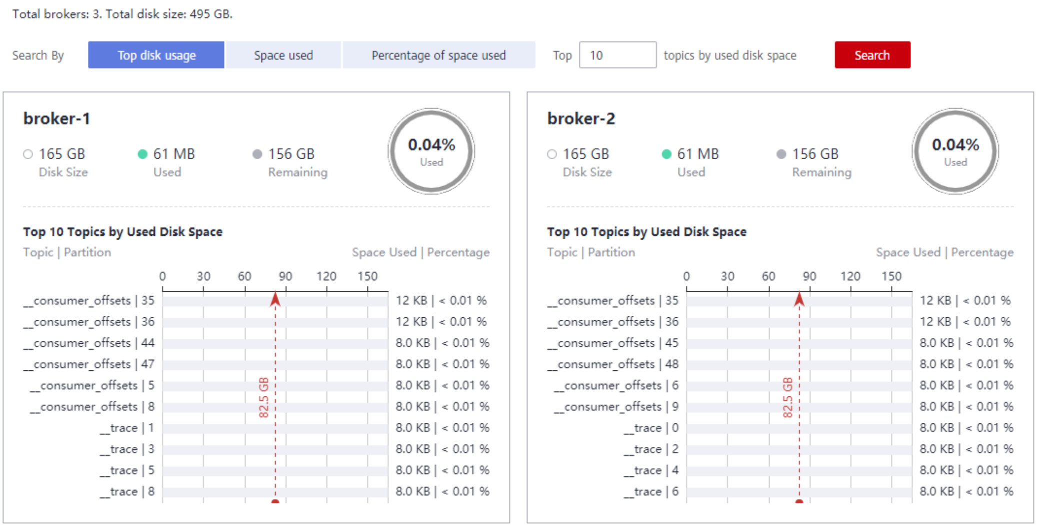
Select the region where your Kafka instance is located.

Step 3 Click and choose **Application** > **Distributed Message Service (for Kafka)** to open the console of DMS for Kafka.

Step 4 Click a Kafka instance to go to the **Basic Information** tab page.

Step 5 Click the **Disk Usage Statistics** tab.

Figure 5-3 Viewing disk usage



Total brokers: 3. Total disk size: 495 GB.

Search By: Top disk usage | Space used | Percentage of space used

Top 10 topics by used disk space

broker-1

○ 165 GB Disk Size ● 61 MB Used ● 156 GB Remaining

Top 10 Topics by Used Disk Space

Topic | Partition	Space Used | Percentage
__consumer_offsets | 35	12 KB | < 0.01 %
__consumer_offsets | 36	12 KB | < 0.01 %
__consumer_offsets | 44	8.0 KB | < 0.01 %
__consumer_offsets | 47	8.0 KB | < 0.01 %
__consumer_offsets | 5	8.0 KB | < 0.01 %
__consumer_offsets | 8	8.0 KB | < 0.01 %
__trace | 1	8.0 KB | < 0.01 %
__trace | 3	8.0 KB | < 0.01 %
__trace | 5	8.0 KB | < 0.01 %
__trace | 8	8.0 KB | < 0.01 %

broker-2

○ 165 GB Disk Size ● 61 MB Used ● 156 GB Remaining

Top 10 Topics by Used Disk Space

Topic | Partition	Space Used | Percentage
__consumer_offsets | 35	12 KB | < 0.01 %
__consumer_offsets | 36	12 KB | < 0.01 %
__consumer_offsets | 45	8.0 KB | < 0.01 %
__consumer_offsets | 48	8.0 KB | < 0.01 %
__consumer_offsets | 6	8.0 KB | < 0.01 %
__consumer_offsets | 9	8.0 KB | < 0.01 %
__trace | 0	8.0 KB | < 0.01 %
__trace | 2	8.0 KB | < 0.01 %
__trace | 4	8.0 KB | < 0.01 %
__trace | 6	8.0 KB | < 0.01 %

You can query topics that use the most disk space or topics that have used a specified amount or percentage of disk space.

In the upper right corner of the page, click **View Metric**. On the displayed Cloud Eye page, you can view metrics of Kafka instances.

----End

6 Managing Topics

6.1 Creating a Topic

A topic is a stream of messages. If automatic topic creation is not enabled during Kafka instance creation, you need to manually create topics for creating and retrieving messages. If automatic topic creation has been enabled for the instance, this operation is optional.

If automatic topic creation is enabled, the system automatically creates a topic when a message is created in or retrieved from a topic that does not exist. This topic has the following default settings: 3 partitions, 3 replicas, aging time of 72 hours, and synchronous replication and flushing disabled. After you change the value of the **log.retention.hours**, **default.replication.factor**, or **num.partitions** parameter, automatically created topics later use the new value. For example, if **num.partitions** is set to **5**, an automatically created topic will have the following settings: 5 partitions, 3 replicas, aging time 72 hours, and synchronous replication and flushing disabled.

There is a limit on the total number of partitions in topics. **When the partition quantity limit is reached, you can no longer create topics.** The total number of partitions varies with instance specifications. For details, see **Specifications**.

Methods that can be used to manually create a topic:

- **Method 1: Creating a Topic on the Console**
- **Method 2: Creating a Topic on Kafka Manager**
- **Method 3: Creating a Topic by Using Kafka CLI**

NOTE

If an instance node is faulty, an internal service error may be reported when you query messages in a topic with only one replica. Therefore, you are not advised using a topic with only one replica.

Method 1: Creating a Topic on the Console

Step 1 Log in to the management console.

Step 2 Click 📍 in the upper left corner to select a region.

NOTE

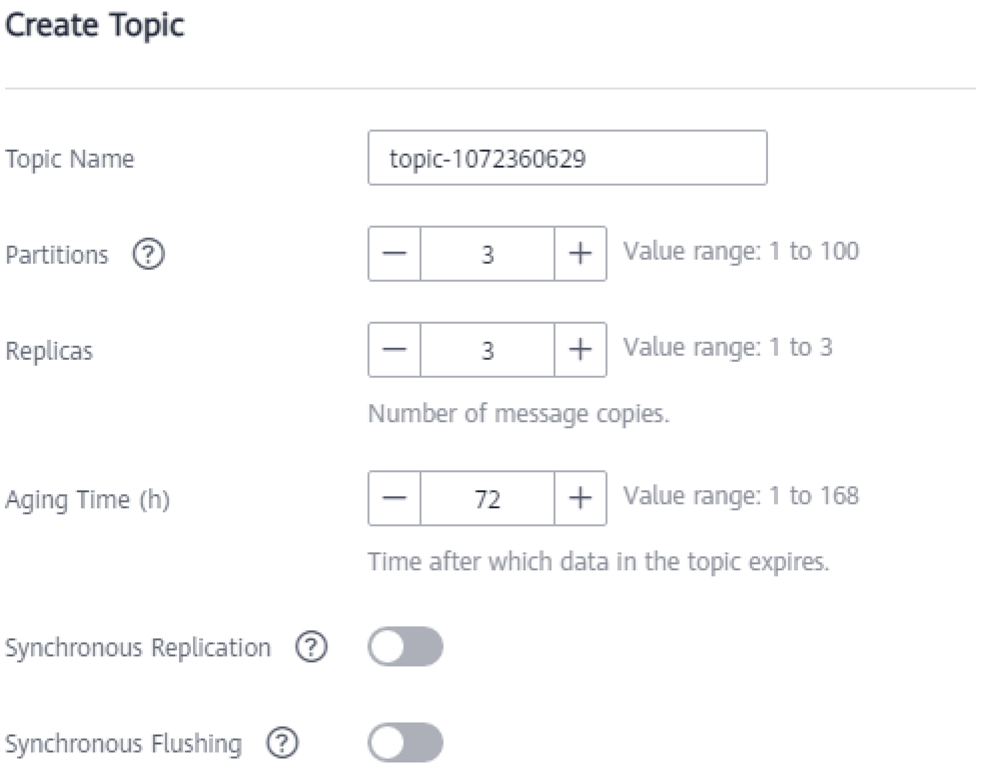
Select the region where your Kafka instance is located.

Step 3 Click ☰ and choose **Application** > **Distributed Message Service (for Kafka)** to open the console of DMS for Kafka.

Step 4 Click the desired Kafka instance to view the instance details.

Step 5 In the navigation pane, choose **Topics**. Then click **Create Topic**.

Figure 6-1 Creating a topic



Create Topic

Topic Name: topic-1072360629

Partitions: 3 (Value range: 1 to 100)

Replicas: 3 (Value range: 1 to 3)
Number of message copies.

Aging Time (h): 72 (Value range: 1 to 168)
Time after which data in the topic expires.

Synchronous Replication: ○ off

Synchronous Flushing: ○ off

Step 6 Specify the topic parameters listed in the following table.

Table 6-1 Topic parameters

Parameter	Description
Topic Name	When creating a topic, you can modify the automatically generated topic name. Once the topic is created, you cannot modify its name.

Parameter	Description
Partitions	A larger number of partitions for a topic indicates more messages retrieved concurrently. If this parameter is set to **1**, messages will be retrieved in the FIFO order. Value range: 1 to 100 Default value: **3**
Replicas	A higher number of replicas delivers higher reliability. Data is automatically backed up on each replica. When one Kafka broker becomes faulty, data is still available on other brokers. If this parameter is set to **1**, only one set of data is available. Default value: **3** **NOTE** If an instance node is faulty, an internal service error may be reported when you query messages in a topic with only one replica. Therefore, you are not advised using a topic with only one replica.
Aging Time (h)	The period that messages are retained for. Consumers must retrieve messages before this period ends. Otherwise, the messages will be deleted and can no longer be retrieved. Value range: 1 to 168 Default value: **72**
Synchronous Replication	A message is returned to the client only after the message creation request has been received and the message has been acknowledged by all replicas. After enabling synchronous replication, set **acks** to **all** or **–1** on the client. Otherwise, this function will not take effect. If there is only one replica, synchronous replication cannot be enabled.
Synchronous Flushing	An indicator of whether a message is immediately flushed to disk once created. ● Enabled: A message is immediately flushed to disk once it is created, resulting in higher reliability. ● Disabled: A message is stored in the memory instead of being immediately flushed to disk once created.

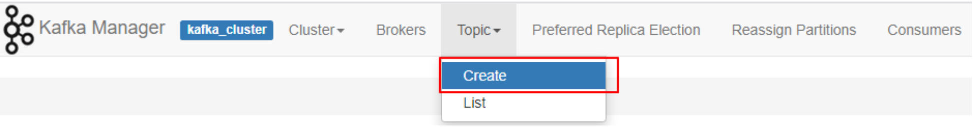
Step 7 Click **OK**.

----End

Method 2: Creating a Topic on Kafka Manager

Log in to Kafka Manager, choose **Topic** > **Create**, and set parameters as prompted.

Figure 6-2 Creating a topic on Kafka Manager



NOTICE

If a topic name starts with a special character, for example, a number sign (#), monitoring data cannot be displayed.

Method 3: Creating a Topic by Using Kafka CLI

If your client is v2.2 or later, you can use **kafka-topics.sh** to create topics and manage topic parameters.

NOTICE

If a topic name starts with a special character, for example, a number sign (#), monitoring data cannot be displayed.

- If SASL is not enabled for the Kafka instance, run the following command in the */{directory where the CLI is located}***/kafka_{version}/bin/** directory to create a topic:
  ```
  ./kafka-topics.sh --create --topic {topic_name} --bootstrap-server {broker_ip}:{port} --partitions {partition_num} --replication-factor {replication_num}
  ```
- If SASL has been enabled for the Kafka instance, perform the following steps to create a topic:

  a. (Optional) If the SSL certificate configuration has been set, skip this step. Otherwise, perform the following operations:

     Create the **ssl-user-config.properties** file in the **/config** directory of the Kafka client and add the SSL certificate configurations by referring to **Step 3**.

  b. Run the following command in the */{directory where the CLI is located}/* **kafka_{version}/bin/** directory to create a topic:
     ```
     ./kafka-topics.sh --create --topic {topic_name} --bootstrap-server {broker_ip}:{port} --partitions {partition_num} --replication-factor {replication_num} --command-config ./config/ssl-user-config.properties
     ```

6.2 Deleting a Topic

Delete a topic using either of the following methods:

- **By using the console**
- **By using Kafka CLI**

Prerequisites

- A Kafka instance has been created, and a topic has been created in this instance.
- The Kafka instance is in the **Running** state.

Deleting a Topic on the Console

Step 1 Log in to the management console.

Step 2 Click 📍 in the upper left corner to select a region.

> **NOTE**
>
> Select the region where your Kafka instance is located.

Step 3 Click ☰ and choose **Application** > **Distributed Message Service (for Kafka)** to open the console of DMS for Kafka.

Step 4 Click the desired Kafka instance to view the instance details.

Step 5 Click the **Topics** tab.

Step 6 Delete topics using either of the following methods:

- Select one or more topics and click **Delete Topic** in the upper left corner.
- In the row containing the topic you want to delete, choose **More** > **Delete**.

Step 7 In the **Delete Topic** dialog box that is displayed, click **Yes** to delete the topic.

----End

Deleting a Topic with the Kafka CLI

If your Kafka client version is later than 2.2, you can use **kafka-topics.sh** to delete topics.

- If SASL is not enabled for the Kafka instance, run the following command in the */{directory where the CLI is located}***/kafka_{version}/bin/** directory to delete a topic:

```
./kafka-topics.sh --bootstrap-server {broker_ip}:{port} --delete --topic {topic_name}
```

- If SASL has been enabled for the Kafka instance, perform the following steps to delete a topic:

  a. (Optional) If the SSL certificate configuration has been set, skip this step. Otherwise, perform the following operations:

     Create the **ssl-user-config.properties** file in the **/config** directory of the Kafka client and add the SSL certificate configurations by referring to **Step 3**.

  b. Run the following command in the */{directory where the CLI is located}***/kafka_{version}/bin/** directory to delete a topic:

```
./kafka-topics.sh --bootstrap-server {broker_ip}:{port} --delete --topic {topic_name} --command-config ./config/ssl-user-config.properties
```

6.3 Modifying Topic Aging Time

Aging time is a period that messages in the topic are retained for. Consumers must retrieve messages before this period ends. Otherwise, the messages will be deleted and can no longer be retrieved.

After creating a topic, you can change its aging time based on service requirements. Changing the aging time does not affect services. The default aging time is 72 hours.

You can change the aging time in either of the following ways:

- By editing the topic on the **Topics** tab page
- By changing the value of the **log.retention.hours** parameter on the **Parameters** tab page. For details, see **Modifying Kafka Parameters**.

NOTE

The **log.retention.hours** parameter takes effect only for topics that have no aging time configured. If there is aging time configured for a topic, it overrides the **log.retention.hours** parameter. For example, if the aging time of Topic01 is set to 60 hours and **log.retention.hours** is set to 72 hours, the actual aging time of Topic01 is 60 hours.

Procedure

Step 1 Log in to the management console.

Step 2 Click 📍 in the upper left corner to select a region.

NOTE

Select the region where your Kafka instance is located.

Step 3 Click ☰ and choose **Application** > **Distributed Message Service (for Kafka)** to open the console of DMS for Kafka.

Step 4 Click the desired Kafka instance to view the instance details.

Step 5 Click the **Topics** tab.

Step 6 Modify the topic aging time using either of the following methods:

- Select one or more topics and click **Edit Topic** in the upper left corner.
- In the row containing the desired topic, click **Edit**.

Step 7 In the **Edit Topic** dialog box, enter the aging time and click **OK**.

----End

6.4 Changing Partition Quantity

After creating a topic, you can increase the number of partitions based on service requirements.

NOTE

Changing the number of partitions does not affect services.

Methods for changing the partition quantity:

- Method 1: By Using the Console
- Method 2: By Using Kafka Manager
- Method 3: By using Kafka CLI

Method 1: By Using the Console

Step 1 Log in to the management console.

Step 2 Click 📍 in the upper left corner to select a region.

NOTE

Select the region where your Kafka instance is located.

Step 3 Click ☰ and choose **Application** > **Distributed Message Service (for Kafka)** to open the console of DMS for Kafka.

Step 4 Click the desired Kafka instance to view the instance details.

Step 5 Click the **Topics** tab.

Step 6 Modify the number of partitions using either of the following methods:

- Select one or more topics and click **Edit Topic** in the upper left corner.
- In the row containing the desired topic, click **Edit**.

Step 7 In the **Edit Topic** dialog box, enter the number of partitions and click **OK**.

NOTE

- The number of partitions can only be increased.
- To ensure performance, the Kafka console allows a maximum of 100 partitions for each topic.
- The total number of partitions of all topics cannot exceed the maximum number of partitions allowed by the instance.

----End

Method 2: By Using Kafka Manager

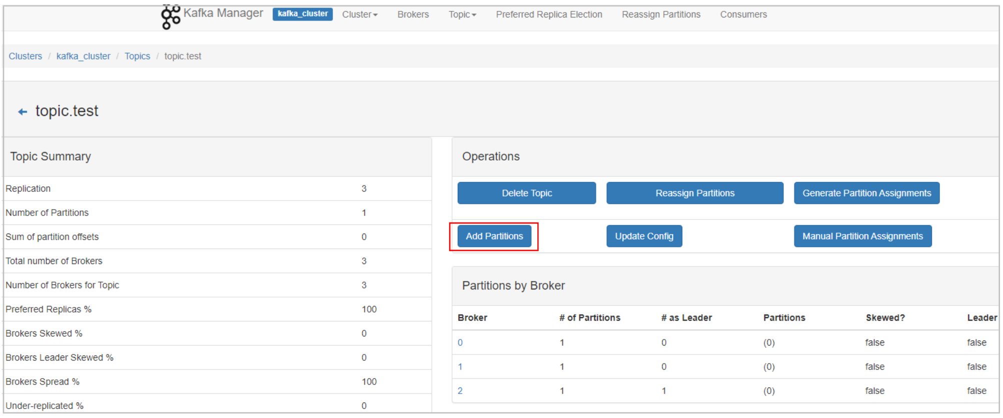
Step 1 Log in to Kafka Manager.

Step 2 Choose **Topic** > **List** to view the list of topics.

Step 3 Click a topic to view its details.

Step 4 Click **Add Partitions**.

Figure 6-3 Topic details page



topic.test

Topic Summary

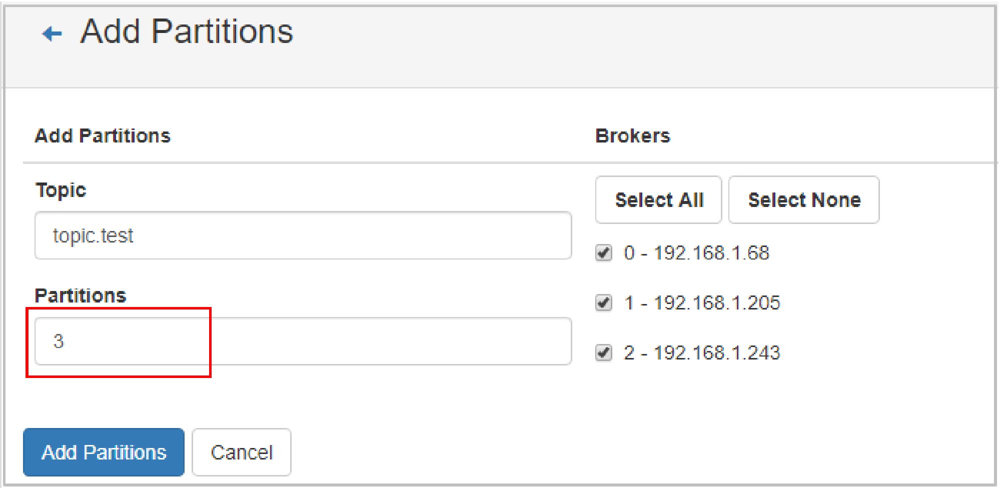
Replication	3
Number of Partitions	1
Sum of partition offsets	0
Total number of Brokers	3
Number of Brokers for Topic	3
Preferred Replicas %	100
Brokers Skewed %	0
Brokers Leader Skewed %	0
Brokers Spread %	100
Under-replicated %	0

Partitions by Broker

Broker	# of Partitions	# as Leader	Partitions	Skewed?	Leader
0	1	0	(0)	false	false
1	1	0	(0)	false	false
2	1	1	(0)	false	false

Step 5 Enter the number of partitions and click **Add Partitions**.

Figure 6-4 Adding partitions



Topic: topic.test

Partitions: 3

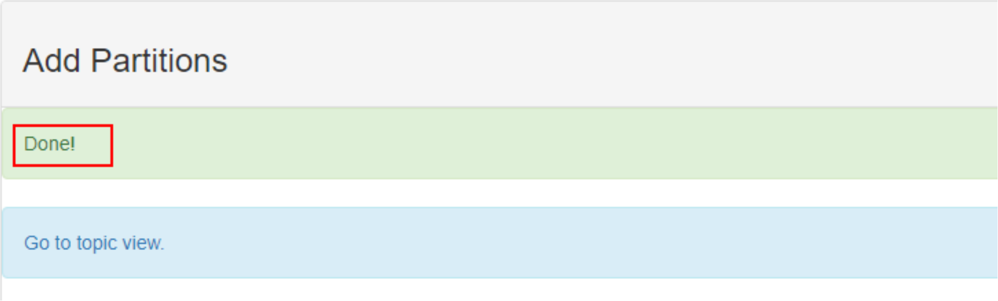
☑ 0 - 192.168.1.68

☑ 1 - 192.168.1.205

☑ 2 - 192.168.1.243

If "Done" is displayed, the partitions are added successfully.

Figure 6-5 Partitions added



Done!

NOTE

- The number of partitions can only be increased.
- The total number of partitions of all topics cannot exceed the maximum number of partitions allowed by the instance.

----End

Method 3: By Using Kafka CLI

If your Kafka client version is later than 2.2, you can use **kafka-topics.sh** to change the partition quantity.

- If SASL is not enabled for the Kafka instance, run the following command in the */{directory where the CLI is located}***/kafka_{version}/bin/** directory to change the partition quantity:
  ```
  ./kafka-topics.sh --bootstrap-server {broker_ip}:{port} --topic {topic_name} --alter --partitions
  {partition_num}
  ```
- If SASL has been enabled for the Kafka instance, perform the following steps to change the partition quantity:
  a. (Optional) If the SSL certificate configuration has been set, skip this step. Otherwise, perform the following operations:

     Create the **ssl-user-config.properties** file in the **/config** directory of the Kafka client and add the SSL certificate configurations by referring to **Step 3**.

  b. Run the following command in the */{directory where the CLI is located}/* **kafka_{version}/bin/** directory to change the partition quantity:
     ```
     ./kafka-topics.sh --bootstrap-server {broker_ip}:{port} --topic {topic_name} --alter --partitions
     {partition_num} --command-config ./config/ssl-user-config.properties
     ```

6.5 Modifying Synchronous Replication and Flushing Settings

Synchronous replication: A message is returned to the client only after the message creation request has been received and the message has been acknowledged by all replicas.

Synchronous flushing: A message is immediately flushed to disk once created.

- Enabled: A message is immediately flushed to disk once it is created, resulting in higher reliability.
- Disabled: A message is stored in the memory instead of being immediately flushed to disk once created.

The following procedure describes how to modify synchronous replication and synchronous flushing settings on the console.

NOTE

Modifying synchronous replication and flushing settings will not restart the instance.

Procedure

Step 1 Log in to the management console.

Step 2 Click 📍 in the upper left corner to select a region.

NOTE

Select the region where your Kafka instance is located.

Step 3 Click ☰ and choose **Application** > **Distributed Message Service (for Kafka)** to open the console of DMS for Kafka.

Step 4 Click the desired Kafka instance to view the instance details.

Step 5 Click the **Topics** tab.

Step 6 Use either of the following methods to modify synchronous replication and synchronous flushing settings:

- Select one or more topics and click **Edit Topic** above the topic list.
- In the row that contains the topic whose synchronous replication and flushing settings are to be modified, click **Edit**.

Step 7 In the **Edit Topic** dialog box, enable or disable synchronous replication and synchronous flushing, and click **OK**.

- To enable them, click ○.
- To disable them, click ●.

NOTE

- If there is only one replica, synchronous replication cannot be enabled.
- After enabling synchronous replication, set **acks** to **all** or **–1** on the client. Otherwise, this function will not take effect.

----End

6.6 Reassigning Partitions

Scenario

Partition reassignment is to reassign replicas of a partition to different brokers to solve the problem of unbalanced broker load.

Partition reassignment is required in the following scenarios:

- After the broker quantity is increased for an instance, the replicas of the original topic partitions are migrated to the new brokers.
- The leader partition is degraded to be a follower on a heavily loaded broker.
- The number of replicas is increased or decreased.

The DMS for Kafka console provides automatic and manual reassignment. Automatic reassignment is recommended because it ensures that leaders are evenly distributed.

Operation Impact

- Partition reassignment on topics with a large amount of data consumes a large amount of network and storage bandwidth. As a result, service requests may time out or the latency may increase. Therefore, you are advised to perform reassignment during off-peak hours.
- A throttle refers to the upper limit of the bandwidth for replication of a topic, to ensure that other topics on the instance are not affected. Note that throttles apply to replication triggered by both normal message production and partition reassignment. If the throttle is too small, normal message production may be affected, and partition reassignment may never complete.
- You cannot delete topics whose reassignment tasks have started. Otherwise, the tasks will never complete.
- You cannot modify the partition quantity of topics whose reassignment tasks have started.
- Reassignment tasks cannot be manually stopped. Please wait until they complete.
- After partition reassignment, the metadata of the topic changes. If the producer does not support the retry mechanism, a few requests will fail, causing some messages to fail to be produced.
- Reassignment takes a long time if the topic has a large amount of data. You are advised to decrease the topic aging time based on the topic consumption so that historical data of the topic can be deleted in a timely manner to accelerate the migration.

Preparing for Partition Reassignment

- To reduce the amount of data to be migrated, decrease the topic aging time without affecting services and wait for messages to age. After the reassignment is complete, you can restore the aging time.
- Ensure that the target broker has sufficient disk capacity. If the remaining disk capacity of the target broker is close to the amount of data to be migrated to the broker, expand the disk capacity before the reassignment.

Auto Reassignment

Step 1 Log in to the management console.

Step 2 Click 📍 in the upper left corner to select a region.

NOTE

Select the region where your Kafka instance is located.

Step 3 Click ☰ and choose **Application** > **Distributed Message Service (for Kafka)** to open the console of DMS for Kafka.

Step 4 Click the desired Kafka instance to view the instance details.

Step 5 In the navigation pane, choose the **Topics** tab.

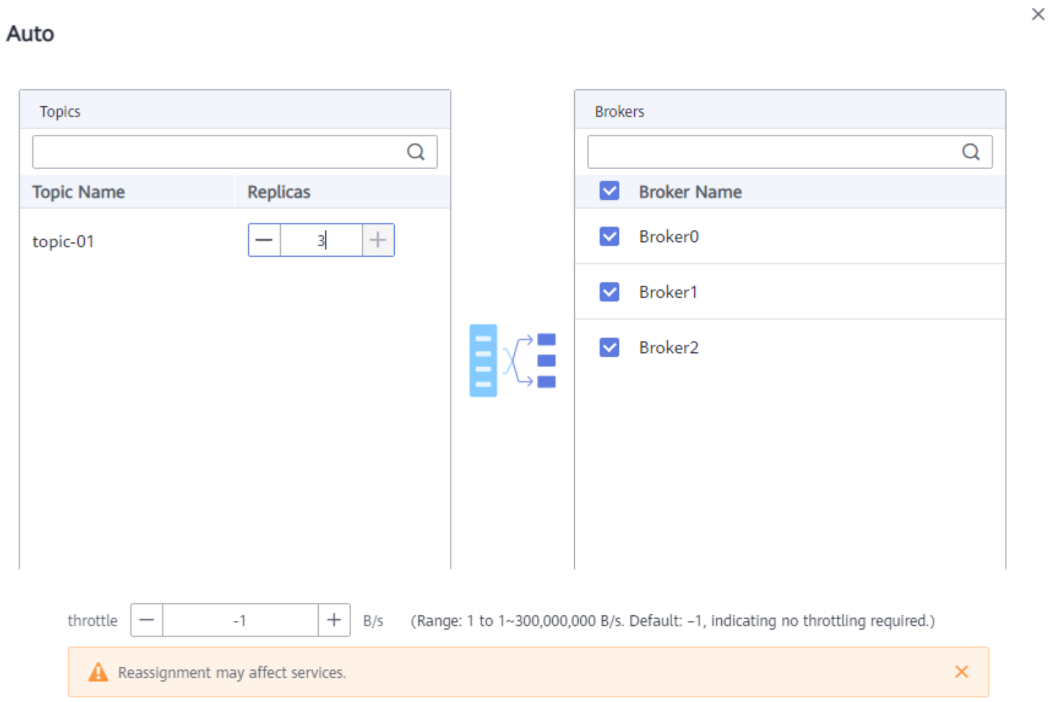
Step 6 Reassign partitions using either of the following methods:

- Select one or more topics and choose **Reassign** > **Auto** above the topic list.
- In the row that contains the desired topic, choose **More** > **Reassign** > **Auto**.

Step 7 Set automatic reassignment parameters.

- In the **Brokers** area, select the brokers to assign the topic's partition replicas to.
- In the **Topics** area, enter the number of replicas to be automatically reassigned. The number of replicas must be less than or equal to the number of brokers.
- Specify **throttle**. The default value is **-1**, indicating that there is no throttle (recommended if the instance load is light). If a throttle is required, you are advised to set it to a value greater than or equal to the total production bandwidth of the to-be-reassigned topic multiplied by the maximum number of replicas of the to-be-reassigned topic. For details, see **Calculating a Throttle**.

Figure 6-6 Setting automatic reassignment parameters



Auto

Topics

Topic Name	Replicas
topic-01	3

Brokers

	Broker Name
☑	Broker0
☑	Broker1
☑	Broker2

throttle -1 B/s (Range: 1 to 1~300,000,000 B/s. Default: –1, indicating no throttling required.)

Reassignment may affect services.

Step 8 Click **OK**. The topic list is displayed.

In the upper left corner of the topic list, click **View details** to view the reassignment task status on the **Background Tasks** page that is displayed. When the task status is **Successful**, reassignment has completed.

NOTE

- You cannot delete topics whose reassignment tasks have started. Otherwise, the tasks will never complete.
- You cannot modify the partition quantity of topics whose reassignment tasks have started.
- Reassignment tasks cannot be manually stopped. Please wait until they complete.

----End

Manual Reassignment

Step 1 Log in to the management console.

Step 2 Click 📍 in the upper left corner to select a region.

NOTE

Select the region where your Kafka instance is located.

Step 3 Click ☰ and choose **Application** > **Distributed Message Service (for Kafka)** to open the console of DMS for Kafka.

Step 4 Click the desired Kafka instance to view the instance details.

Step 5 In the navigation pane, choose the **Topics** tab.

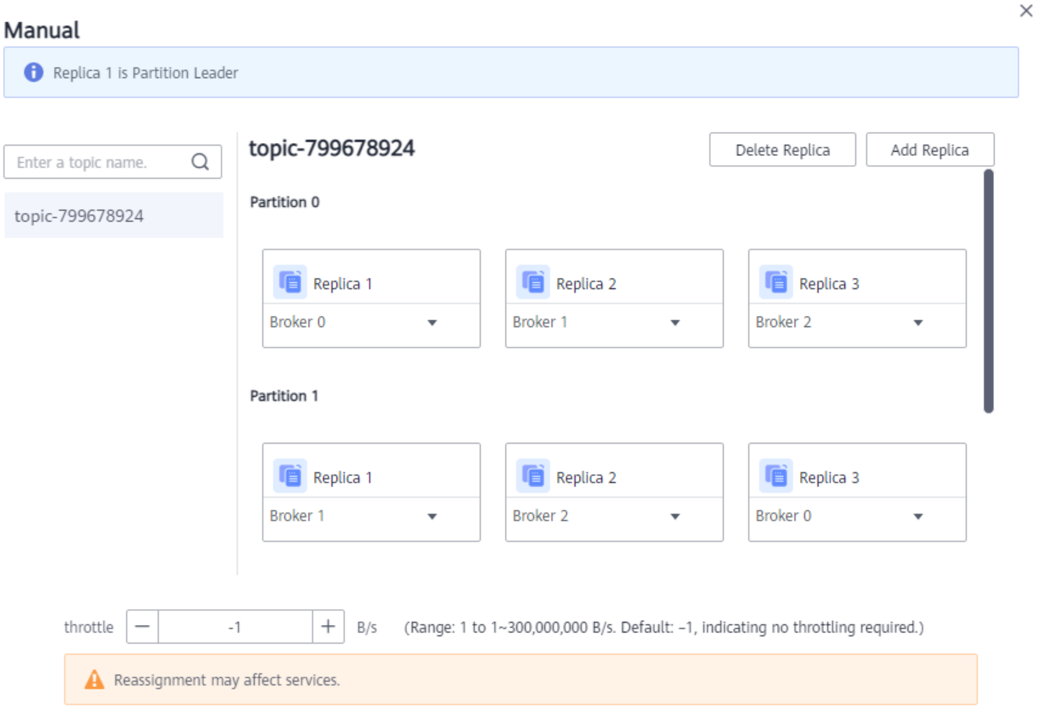
Step 6 Reassign partitions using either of the following methods:

- Select a topic and choose **Reassign** > **Manual** above the topic list. Manual reassignment does not support batch operations.
- In the row that contains the desired topic, choose **More** > **Reassign** > **Manual**.

Step 7 Set manual reassignment parameters.

- In the upper right corner of the **Manual** dialog box, click **Delete Replica** or **Add Replica** to reduce or increase the number of replicas for each partition of the topic.
- Under the name of the replica to be reassigned, click the broker name or ▾ and select the target broker to migrate the replica to. Assign replicas of the same partition to different brokers.
- Specify **throttle**. The default value is **-1**, indicating that there is no throttle (recommended if the instance load is light). If a throttle is required, you are advised to set it to a value greater than or equal to the total production bandwidth of the to-be-reassigned topic multiplied by the maximum number of replicas of the to-be-reassigned topic. For details, see **Calculating a Throttle**.

Figure 6-7 Setting manual reassignment parameters



Replica 1 is Partition Leader

topic-799678924

topic-799678924

Partition 0

Replica 1: Broker 0

Replica 2: Broker 1

Replica 3: Broker 2

Partition 1

Replica 1: Broker 1

Replica 2: Broker 2

Replica 3: Broker 0

throttle: -1 B/s (Range: 1 to 1~300,000,000 B/s. Default: -1, indicating no throttling required.)

Reassignment may affect services.

Step 8 Click **OK**. The topic list is displayed.

In the upper left corner of the topic list, click **View details** to view the reassignment task status on the **Background Tasks** page that is displayed. When the task status is **Successful**, reassignment has completed.

NOTE

- You cannot delete topics whose reassignment tasks have started. Otherwise, the tasks will never complete.
- You cannot modify the partition quantity of topics whose reassignment tasks have started.
- Reassignment tasks cannot be manually stopped. Please wait until they complete.

----End

Calculating a Throttle

Throttles are affected by the execution duration of the reassignment, leader/follower distribution of partition replicas, and message production rate.

- A throttle limits the replication traffic of all partitions in a broker.
- Replicas added after the assignment are regarded as followers, and existing replicas are regarded as leaders. Throttles on leaders and followers are separated.
- Throttles do not distinguish between replication caused by normal message production and that caused by partition reassignment. Therefore, the traffic generated in both cases is throttled.

Assume that the partition reassignment task needs to be completed within 200s and each replica has 100 MB data. Calculate the throttle in the following scenarios:

Scenario 1: Topic 1 has two partitions and two replicas, and Topic 2 has one partition and one replica. All leader replicas are on the same broker. One replica needs to be added for Topic 1 and Topic 2 respectively.

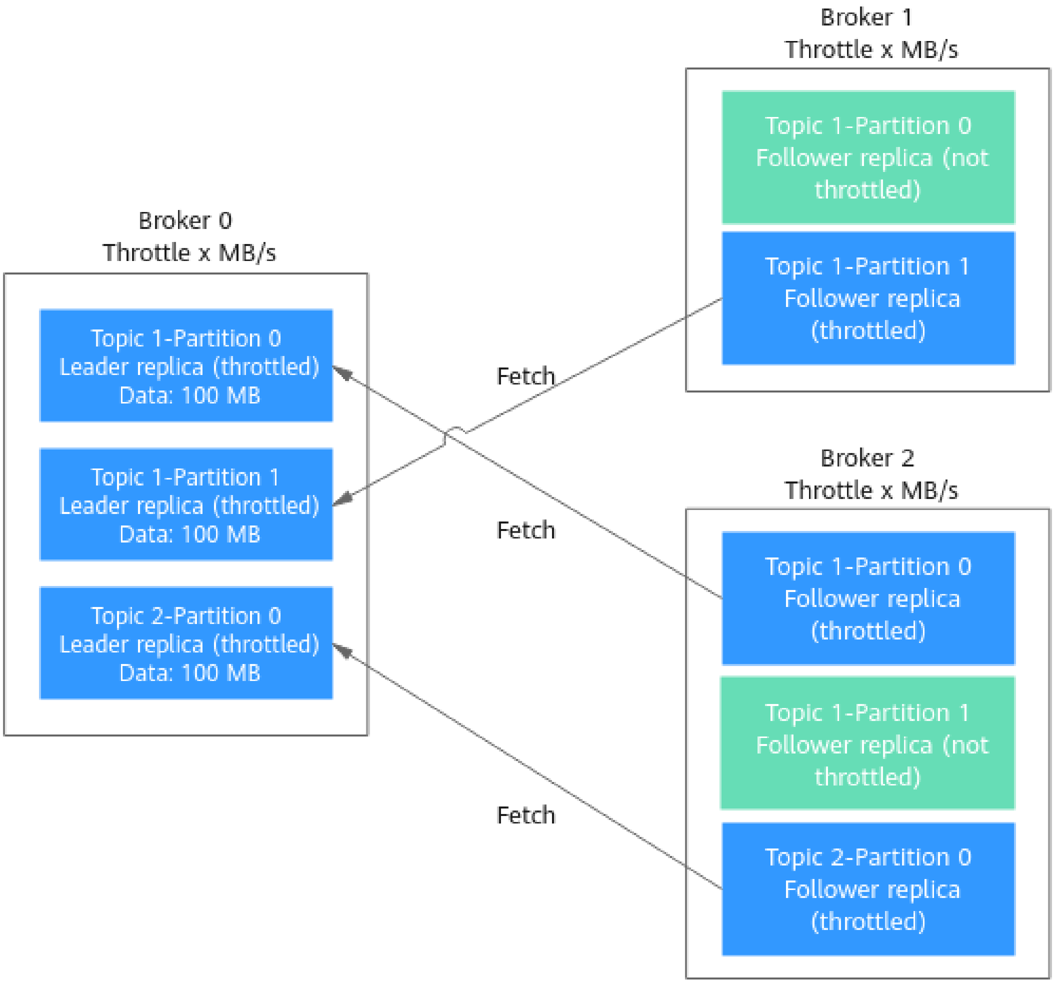
Table 6-2 Replica distribution before reassignment

Topic Name	Partition Name	Broker of Leader Replica	Broker of Follower Replica
Topic 1	0	0	0, 1
Topic 1	1	0	0, 2
Topic 2	0	0	0

Table 6-3 Replica distribution after reassignment

Topic Name	Partition Name	Broker of Leader Replica	Broker of Follower Replica
Topic 1	0	0	0, 1, 2
Topic 1	1	0	0, 1, 2
Topic 2	0	0	0, 2

Figure 6-8 Reassignment scenario 1


Broker 1
Throttle x MB/s
Topic 1-Partition 0 Follower replica (not throttled)
Topic 1-Partition 1 Follower replica (throttled)
Broker 0
Throttle x MB/s
Topic 1-Partition 0 Leader replica (throttled) Data: 100 MB
Topic 1-Partition 1 Leader replica (throttled) Data: 100 MB
Topic 2-Partition 0 Leader replica (throttled) Data: 100 MB
Fetch
Fetch
Fetch
Broker 2
Throttle x MB/s
Topic 1-Partition 0 Follower replica (throttled)
Topic 1-Partition 1 Follower replica (not throttled)
Topic 2-Partition 0 Follower replica (throttled)

As shown in **Figure 6-8**, three replicas fetch data from Broker 0. Each replica on Broker 0 has 100 MB data. Broker 0 has only leader replicas, and Broker 1 and Broker 2 have only follower replicas.

- Bandwidth required by Broker 0 to complete partition reassignment within 200s = (100 MB + 100 MB + 100 MB)/200s = 1.5 MB/s
- Bandwidth required by Broker 1 to complete partition reassignment within 200s = 100 MB/200s = 0.5 MB/s
- Bandwidth required by Broker 2 to complete partition reassignment within 200s = (100 MB + 100 MB)/200s = 1 MB/s

In conclusion, to complete the partition reassignment task within 200s, set the throttle to a value greater than or equal to 1.5 MB/s.

Scenario 2: Topic 1 has two partitions and one replica, and Topic 2 has two partitions and one replica. Leader replicas are on different brokers. One replica needs to be added for Topic 1 and Topic 2 respectively.

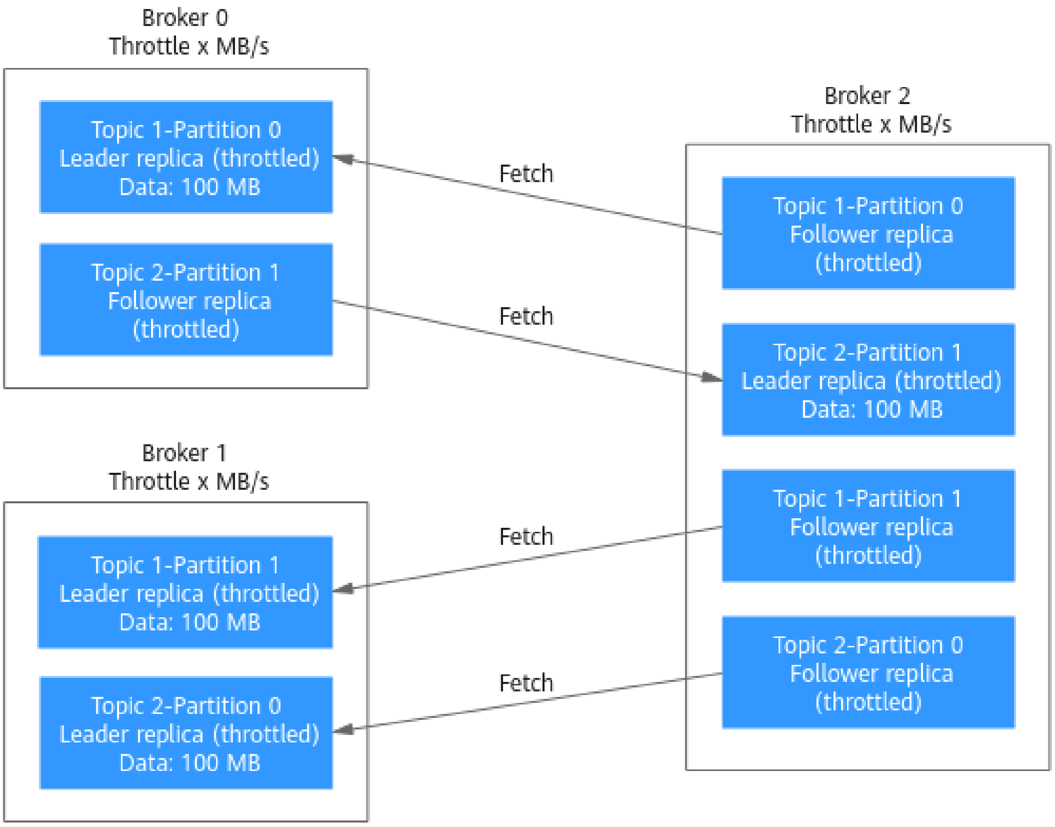
Table 6-4 Replica distribution before reassignment

Topic Name	Partition Name	Broker of Leader Replica	Broker of Follower Replica
Topic 1	0	0	0
Topic 1	1	1	1
Topic 2	0	1	1
Topic 2	1	2	2

Table 6-5 Replica distribution after reassignment

Topic Name	Partition Name	Broker of Leader Replica	Broker of Follower Replica
Topic 1	0	0	0, 2
Topic 1	1	1	1, 2
Topic 2	0	1	1, 2
Topic 2	1	2	2, 0

Figure 6-9 Reassignment scenario 2



As shown in **Figure 6-9**, Broker 1 has only leader replicas, and Broker 0 and Broker 2 have both leader and follower replicas. Leader and follower replicas on Broker 0 and Broker 2 are throttled separately.

- Bandwidth required by Broker 0 (leader) to complete partition reassignment within 200s = 100 MB/200s = 0.5 MB/s
- Bandwidth required by Broker 0 (follower) to complete partition reassignment within 200s = 100 MB/200s = 0.5 MB/s
- Bandwidth required by Broker 1 to complete partition reassignment within 200s = (100 MB + 100 MB)/200s = 1 MB/s
- Bandwidth required by Broker 2 (leader) to complete partition reassignment within 200s = 100 MB/200s = 0.5 MB/s
- Bandwidth required by Broker 2 (follower) to complete partition reassignment within 200s = (100 MB + 100 MB + 100 MB)/200s = 1.5 MB/s

In conclusion, to complete the partition reassignment task within 200s, set the throttle to a value greater than or equal to 1.5 MB/s.

Scenario 3: Both Topic 1 and Topic 2 have one partition and two replicas. All leader replicas are on the same broker. One replica needs to be added to Topic 1. Messages are produced on Topic 1, causing replication.

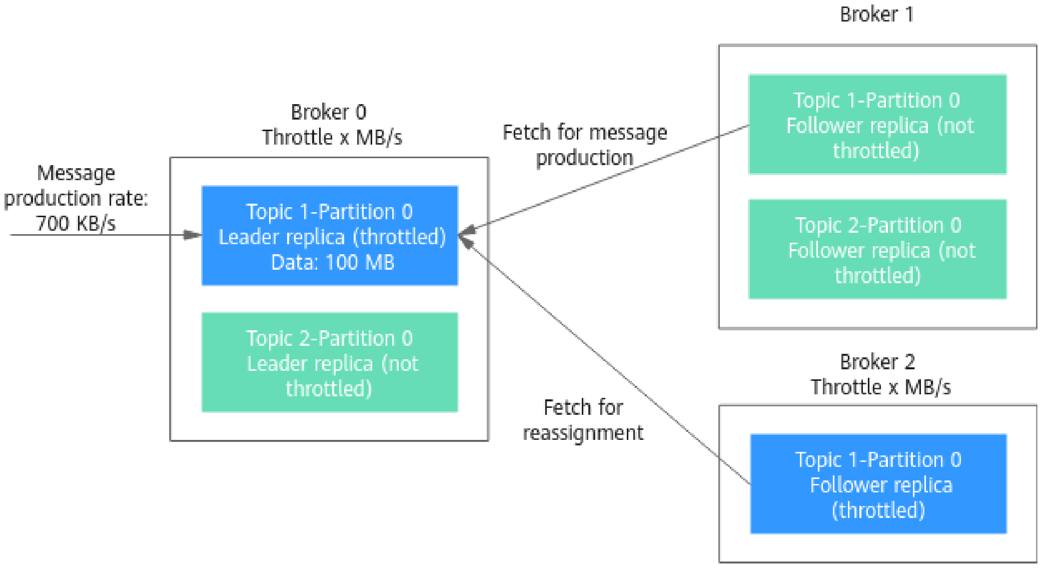
Table 6-6 Replica distribution before reassignment

Topic Name	Partition Name	Broker of Leader Replica	Broker of Follower Replica
Topic 1	0	0	0, 1
Topic 2	0	0	0, 1

Table 6-7 Replica distribution after reassignment

Topic Name	Partition Name	Broker of Leader Replica	Broker of Follower Replica
Topic 1	0	0	0, 1, 2
Topic 2	0	0	0, 1

Figure 6-10 Reassignment scenario 3


Broker 1
Topic 1-Partition 0 Follower replica (not throttled)
Topic 2-Partition 0 Follower replica (not throttled)
Broker 0 Throttle x MB/s
Fetch for message production
Message production rate: 700 KB/s
Topic 1-Partition 0 Leader replica (throttled) Data: 100 MB
Topic 2-Partition 0 Leader replica (not throttled)
Broker 2 Throttle x MB/s
Fetch for reassignment
Topic 1-Partition 0 Follower replica (throttled)

As shown in **Figure 6-10**, one replica needs to fetch data from Broker 0 for partition reassignment, and the other replica needs to fetch data from Broker 0 for message production. Since the throttle does not distinguish between message production and partition reassignment, the traffic caused by both is limited and counted.

- Bandwidth required by Broker 0 to complete partition reassignment within 200s = (100 MB + 700 KB/s x 200s)/200s + 700 KB/s= 1.9 MB/s
- Bandwidth required by Broker 2 to complete partition reassignment within 200s = 100 MB/200s = 0.5 MB/s

In conclusion, to complete the partition reassignment task within 200s, set the throttle to a value greater than or equal to 1.9 MB/s.

6.7 Viewing Sample Code

On the console, view sample code for creating and retrieving messages in Java, Go, and Python.

Procedure

Step 1 Log in to the management console.

Step 2 Click ⊙ in the upper left corner to select a region.

NOTE

Select the region where your Kafka instance is located.

Step 3 Click ☰ and choose **Application** > **Distributed Message Service (for Kafka)** to open the console of DMS for Kafka.

Step 4 Click the desired Kafka instance to view the instance details.

Step 5 Click the **Topics** tab.

Step 6 Click **View Sample Code**. The **Sample Code** dialog box is displayed.

View sample code for creating and retrieving messages in Java, Go, and Python. Set **Access By** to **PlainText** to view the sample code where SASL_SSL authentication is disabled. Set **Access By** to **SASL_SSL** to view the sample code where SASL_SSL authentication is enabled.

----End

6.8 Exporting Topics

Export topics on the console. Batch export is supported.

Prerequisites

A topic has been created.

Procedure

Step 1 Log in to the management console.

Step 2 Click 📍 in the upper left corner to select a region.

NOTE

Select the region where your Kafka instance is located.

Step 3 Click ☰ and choose **Application** > **Distributed Message Service (for Kafka)** to open the console of DMS for Kafka.

Step 4 Click the desired Kafka instance to view the instance details.

Step 5 Click the **Topics** tab.

Step 6 Click ↗ in the upper right to export the topic list.

The topic list contains the following information: topic name, number of partitions, number of replicas, aging time, and whether synchronous replication and flushing are enabled.

----End

6.9 Configuring Topic Permissions

DMS for Kafka supports ACL permission management for topics. You can differentiate the operations that different users are allowed to perform on a topic by granting the users different permissions.

This section describes how to grant topic permissions to a SASL_SSL user after SASL_SSL is enabled. For details about how to create a SASL_SSL user, see **Creating a SASL_SSL User**.

Constraints

- If no SASL_SSL user is granted any permission for a topic, all users can subscribe to or publish messages to the topic.
- If one or more SASL_SSL users are granted permissions for a topic, only the authorized users can subscribe to or publish messages to the topic.

Prerequisites

- SASL_SSL has been enabled when you create the Kafka instance.
- (Optional) A SASL_SSL user has been created. For details, see **Creating a SASL_SSL User**.

Configuring Topic Permissions

Step 1 Log in to the management console.

Step 2 Click 📍 in the upper left corner to select a region.

NOTE

Select the region where your Kafka instance is located.

Step 3 Click ☰ and choose **Application** > **Distributed Message Service (for Kafka)** to open the console of DMS for Kafka.

Step 4 Click the desired Kafka instance to view the instance details.

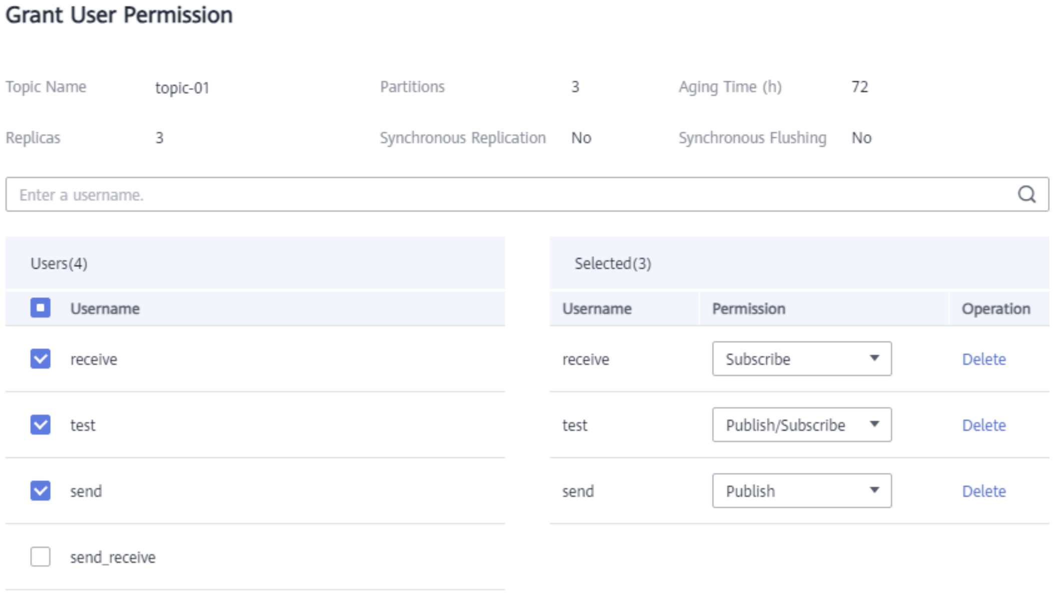
Step 5 In the navigation pane, choose the **Topics** tab.

Step 6 In the row that contains the topic for which you want to configure user permissions, click **Grant User Permission**.

In the upper part of the **Grant User Permission** dialog box, the topic information is displayed, including the topic name, number of partitions, aging time, number of replicas, and whether synchronous flushing and replication are enabled. In the middle part, you can use the search box to search for a user if there are many SASL_SSL users. In the **Users** area, the list of created SASL_SSL users is displayed. In the **Selected** area, you can grant permissions to the SASL_SSL users.

Step 7 In the **Users** area of the **Grant User Permission** dialog box, select target users. In the **Selected** area, configure permissions (**Subscribe**, **Publish**, or **Publish/Subscribe**) for the users.

Figure 6-11 Granting user permissions



Grant User Permission

Topic Name	topic-01	Partitions	3	Aging Time (h)	72
Replicas	3	Synchronous Replication	No	Synchronous Flushing	No

Enter a username.

Users(4)

■	Username
☑	receive
☑	test
☑	send
□	send_receive

Selected(3)

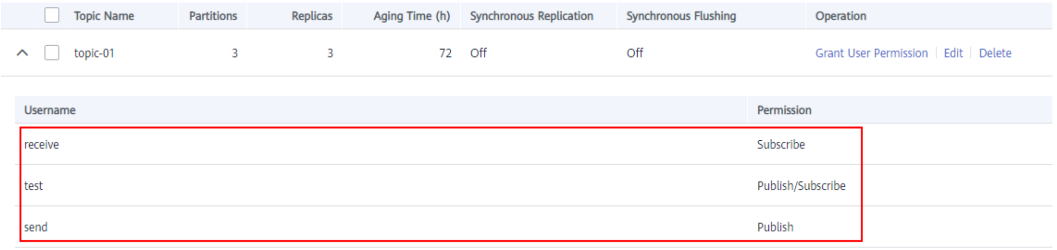
Username	Permission	Operation
receive	Subscribe	Delete
test	Publish/Subscribe	Delete
send	Publish	Delete

As shown in **Figure 6-11**, only the **test**, **send**, and **receive** users can subscribe to or publish messages to topic-01. The **send_receive** user cannot subscribe to or publish messages to topic-01.

Step 8 Click **OK**.

On the **Topics** tab page, click ∨ next to the topic name to view the authorized users and their permissions.

Figure 6-12 Viewing authorized users and their permissions



□	Topic Name	Partitions	Replicas	Aging Time (h)	Synchronous Replication	Synchronous Flushing	Operation
□	topic-01	3	3	72	Off	Off	Grant User Permission | Edit | Delete

Username	Permission
receive	Subscribe
test	Publish/Subscribe
send	Publish

----End

(Optional) Deleting Topic Permissions

Step 1 Log in to the management console.

Step 2 Click 📍 in the upper left corner to select a region.

NOTE

Select the region where your Kafka instance is located.

Step 3 Click ☰ and choose **Application** > **Distributed Message Service (for Kafka)** to open the console of DMS for Kafka.

Step 4 Click the desired Kafka instance to view the instance details.

Step 5 In the navigation pane, choose the **Topics** tab.

Step 6 In the row that contains the topic for which you want to remove user permissions, click **Grant User Permission**.

Step 7 In the **Selected** area of the displayed **Grant User Permission** dialog box, locate the row that contains the SASL_SSL user whose permissions are to be removed, click **Delete**, and click **OK**.

----End

7 Managing Messages

7.1 Querying Messages

Scenario

You can view the offset of different partitions, the message size, creation time, and body of messages in topics.

Procedure

Step 1 Log in to the management console.

Step 2 Click ⚲ in the upper left corner to select a region.

NOTE

Select the region where your Kafka instance is located.

Step 3 Click ☰ and choose **Application** > **Distributed Message Service (for Kafka)** to open the console of DMS for Kafka.

Step 4 Click the desired Kafka instance to view the instance details.

Step 5 Click the **Message Query** tab. Then specify the topic name, partition, and the search method.

If no partition is specified, messages in all partitions of the topic are displayed.

You can search by the following methods:

- **Creation time**: Search by the time that messages are created.
- **Offset**: Search by the message position.

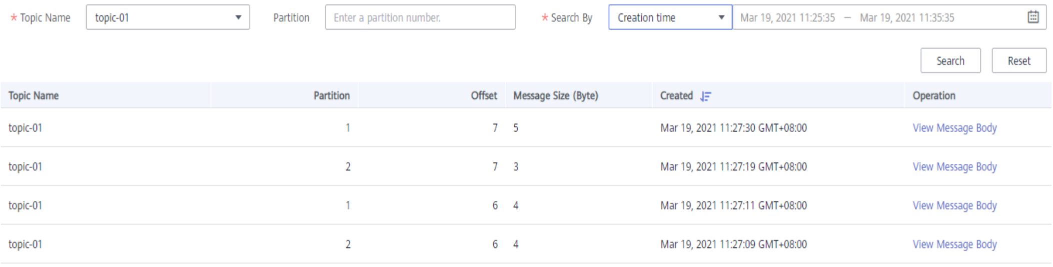
NOTE

If a topic contains a large amount of data, an internal service error may be reported when you query messages in a topic with only one replica. You can shorten the time range for query based on the data volume.

Step 6 Click **Search** to query messages.

The query result is as follows.

Figure 7-1 Querying topic messages



* Topic Name: topic-01

Partition: Enter a partition number.

* Search By: Creation time Mar 19, 2021 11:25:35 – Mar 19, 2021 11:35:35

Topic Name	Partition	Offset	Message Size (Byte)	Created	Operation
topic-01	1	7	5	Mar 19, 2021 11:27:30 GMT+08:00	View Message Body
topic-01	2	7	3	Mar 19, 2021 11:27:19 GMT+08:00	View Message Body
topic-01	1	6	4	Mar 19, 2021 11:27:11 GMT+08:00	View Message Body
topic-01	2	6	4	Mar 19, 2021 11:27:09 GMT+08:00	View Message Body

Parameter description:

- **Topic Name**: name of the topic where the message is located
- **Partition**: partition where the message is located
- **Offset**: position of the message in the partition
- **Message Size (Byte)** size of the message
- **Created**: time when the message is created. The message creation time is specified by **CreateTime** when a producer creates messages. If this parameter is not set during message creation, the message creation time is year 1970 by default.

Step 7 Click **View Message Body**. In the displayed **View Message Body** dialog box, view the message content, including the topic name, partition, offset, creation time, and message body.

NOTE

The console displays messages smaller than 4 KB. To view messages larger than 4 KB, click **Download Message**.

Step 8 (Optional) To restore the default settings, click **Reset**.

----End

8 Managing Users

8.1 Creating a SASL_SSL User

DMS for Kafka supports ACL permission management for topics. You can differentiate the operations that different users are allowed to perform on a topic by granting the users different permissions.

This section describes how to create a SASL_SSL user after SASL_SSL is enabled for a Kafka instance. For details about how to grant user permissions, see **Configuring Topic Permissions**.

A maximum of 20 users can be created for a Kafka instance.

Prerequisites

SASL_SSL has been enabled when you create the Kafka instance.

Procedure

Step 1 Log in to the management console.

Step 2 Click 📍 in the upper left corner to select a region.

NOTE

Select the region where your Kafka instance is located.

Step 3 Click ☰ and choose **Application** > **Distributed Message Service (for Kafka)** to open the console of DMS for Kafka.

Step 4 Click the desired Kafka instance to view the instance details.

Step 5 On the **Users** tab page, click **Create User**.

Step 6 In the displayed **Create User** dialog box, set the username and password, and click OK.

After the SASL_SSL user is created, grant permissions to the user by referring to **Configuring Topic Permissions**.

----End

8.2 Resetting the SASL_SSL Password

Scenario

For a Kafka instance with SASL_SSL enabled, there are two ways to create an SASL_SSL user on the console. Accordingly, there are two ways to reset the SASL_SSL user's password:

- If an SASL_SSL user is created on the **Users** tab page, reset their password by referring to the following instructions.
- If an SASL_SSL user is created during instance creation, reset their password by referring to **Resetting Kafka Password**.

Prerequisites

- You can reset the SASL_SSL password only if Kafka SASL_SSL has been enabled for the instance.
- You can reset the SASL_SSL password only when the instance is in the **Running** state.

Procedure

Step 1 Log in to the management console.

Step 2 Click 📍 in the upper left corner to select a region.

NOTE

Select the region where your Kafka instance is located.

Step 3 Click ☰ and choose **Application** > **Distributed Message Service (for Kafka)** to open the console of DMS for Kafka.

Step 4 Click the name of the desired Kafka instance.

Step 5 On the **Users** tab page, click **Reset Password** in the row containing the desired user.

Step 6 Enter and confirm a new password, and click **OK**.

- If the password is successfully reset, a success message is displayed.
- If the password fails to be reset, a failure message is displayed. In this case, reset the password again. If you still fail to reset the password after multiple attempts, contact customer service.

NOTE

The system will display a success message only after the password is successfully reset on all brokers.

----End

8.3 Deleting a SASL_SSL User

This section describes how to delete a SASL_SSL user.

Procedure

Step 1 Log in to the management console.

Step 2 Click 📍 in the upper left corner to select a region.

NOTE

Select the region where your Kafka instance is located.

Step 3 Click ☰ and choose **Application** > **Distributed Message Service (for Kafka)** to open the console of DMS for Kafka.

Step 4 Click the desired Kafka instance to view its details.

Step 5 Delete a SASL_SSL user using either of the following methods:

- On the **Users** tab page, click **Delete** in the row that contains the SASL_SSL user to be deleted.
- On the **Users** tab page, select one or more SASL_SSL users and click **Delete** above the list.

NOTE

The SASL_SSL user configured during the creation of a Kafka instance cannot be deleted.

Step 6 In the displayed **Delete User** dialog box, click **Yes** to delete the SASL_SSL user.

----End

9 Managing Consumer Groups

9.1 Querying Consumer Group Details

View the consumer group list, consumer list, and consumer offsets.

Prerequisites

The consumer list can be viewed only when consumers in a consumer group are connected to the Kafka instance (that is, the consumer group is in the **STABLE** state).

Viewing the Consumer Group List (Console)

Step 1 Log in to the management console.

Step 2 Click 📍 in the upper left corner to select a region.

NOTE

Select the region where your Kafka instance is located.

Step 3 Click ☰ and choose **Application** > **Distributed Message Service (for Kafka)** to open the console of DMS for Kafka.

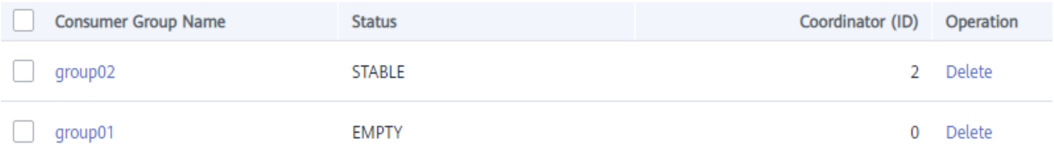
Step 4 Click the desired Kafka instance to view its details.

Step 5 In the navigation pane, choose the **Consumer Groups** tab.

The consumer group name, status, and Coordinator (ID) are displayed. **Coordinator (ID)** indicates the broker where the coordinator component is located. The consumer group status can be:

- **DEAD**: The consumer group has no member or metadata.
- **EMPTY**: The consumer group has metadata but has no member.
- **PREPARING_REBALANCE**: The consumer group is to be rebalanced.
- **COMPLETING_REBALANCE**: All members have joined the consumer group.
- **STABLE**: Members in the consumer group can consume messages normally.

Figure 9-1 Consumer group list



□	Consumer Group Name	Status	Coordinator (ID)	Operation
□	group02	STABLE	2	Delete
□	group01	EMPTY	0	Delete

Step 6 (Optional) To query a specific consumer group, enter the consumer group name in the search box and click 🔍.

Step 7 (Optional) To refresh the consumer group list, click ⟳ in the upper right corner.

----End

Viewing the Consumer Group List (Kafka CLI)

- If SASL is not enabled for the Kafka instance, run the following command in the */{directory where the CLI is located}***/kafka_{version}/bin/** directory to query the consumer group list:

```
./kafka-consumer-groups.sh --bootstrap-server {broker_ip}:{port} --list
```

- If SASL has been enabled for the Kafka instance, perform the following steps to query the consumer group list:
  a. (Optional) If the SSL certificate configuration has been set, skip this step. Otherwise, perform the following operations:

     Create the **ssl-user-config.properties** file in the **/config** directory of the Kafka client and add the SSL certificate configurations by referring to **Step 3**.

  b. Run the following command in the */{directory where the CLI is located}***/kafka_{version}/bin/** directory to query the consumer group list:

```
./kafka-consumer-groups.sh --bootstrap-server {broker_ip}:{port} --list --command-config ./
config/ssl-user-config.properties
```

Viewing the Consumer List (Console)

Step 1 Log in to the management console.

Step 2 Click 📍 in the upper left corner to select a region.

NOTE

Select the region where your Kafka instance is located.

Step 3 Click ☰ and choose **Application** > **Distributed Message Service (for Kafka)** to open the console of DMS for Kafka.

Step 4 Click the desired Kafka instance to view its details.

Step 5 In the navigation pane, choose the **Consumer Groups** tab.

Step 6 Click the name of the desired consumer group.

Step 7 On the **Consumers** tab page, view the consumer list.

In the consumer list, you can view the consumer ID, consumer address, and client ID.

Step 8 (Optional) To query a specific consumer, enter the consumer ID in the search box and click 🔍.

----End

Viewing the Consumer List (Kafka CLI)

- If SASL is not enabled for the Kafka instance, run the following command in the **/***{directory where the CLI is located}***/kafka_{version}/bin/** directory to query the consumer list:

```
./kafka-consumer-groups.sh --bootstrap-server {broker_ip}:{port} --group {group_name} --members --describe
```

- If SASL has been enabled for the Kafka instance, perform the following steps to query the consumer list:

  a. (Optional) If the SSL certificate configuration has been set, skip this step. Otherwise, perform the following operations:

  Create the **ssl-user-config.properties** file in the **/config** directory of the Kafka client and add the SSL certificate configurations by referring to **Step 3**.

  b. Run the following command in the **/***{directory where the CLI is located}***/ kafka_{version}/bin/** directory to query the consumer list:

```
./kafka-consumer-groups.sh --bootstrap-server {broker_ip}:{port} --group {group_name} --members --describe --command-config ./config/ssl-user-config.properties
```

Viewing Consumer Offsets (Console)

Step 1 Log in to the management console.

Step 2 Click 📍 in the upper left corner to select a region.

NOTE

Select the region where your Kafka instance is located.

Step 3 Click ☰ and choose **Application** > **Distributed Message Service (for Kafka)** to open the console of DMS for Kafka.

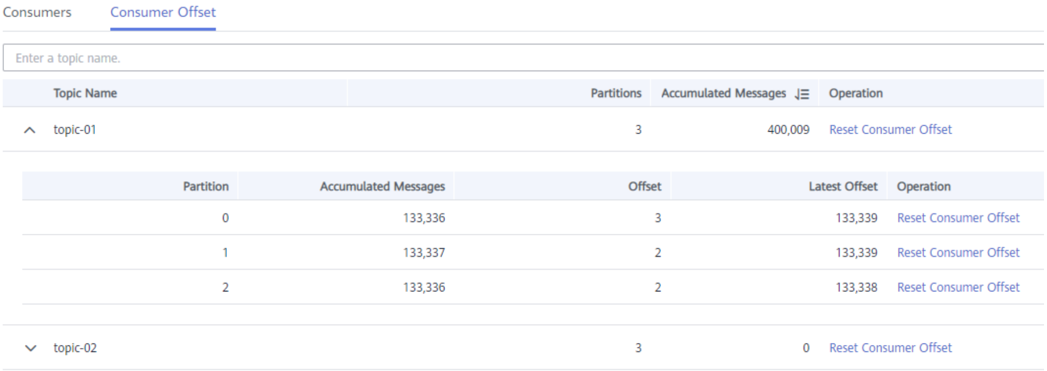
Step 4 Click the desired Kafka instance to view its details.

Step 5 In the navigation pane, choose the **Consumer Groups** tab.

Step 6 Click the name of the desired consumer group.

Step 7 On the **Consumer Offset** tab page, view the list of topics that the consumer group has subscribed to, total number of messages accumulated in the topic, number of messages accumulated in each partition of the topic, offset of each partition, and latest offset.

Figure 9-2 Consumer offsets



Topic Name	Partitions	Accumulated Messages	Operation
topic-01	3	400,009	Reset Consumer Offset

Partition	Accumulated Messages	Offset	Latest Offset	Operation
0	133,336	3	133,339	Reset Consumer Offset
1	133,337	2	133,339	Reset Consumer Offset
2	133,336	2	133,338	Reset Consumer Offset

Topic Name	Partitions	Accumulated Messages	Operation
topic-02	3	0	Reset Consumer Offset

Step 8 (Optional) To query the consumer offsets of a specific topic, enter the topic name in the search box and click 🔍.

----End

Viewing Consumer Offsets (Kafka CLI)

- If SASL is not enabled for the Kafka instance, run the following command in the **/***{directory where the CLI is located}***/kafka_{version}/bin/** directory to query consumer offsets:

```
./kafka-consumer-groups.sh --bootstrap-server {broker_ip}:{port} --offsets --describe --all-groups
```

- If SASL has been enabled for the Kafka instance, perform the following steps to query consumer offsets:

  a. (Optional) If the SSL certificate configuration has been set, skip this step. Otherwise, perform the following operations:

     Create the **ssl-user-config.properties** file in the **/config** directory of the Kafka client and add the SSL certificate configurations by referring to **Step 3**.

  b. Run the following command in the **/***{directory where the CLI is located}***/kafka_{version}/bin/** directory to query consumer offsets:

```
./kafka-consumer-groups.sh --bootstrap-server {broker_ip}:{port} --offsets --describe --all-
groups --command-config ./config/ssl-user-config.properties
```

9.2 Deleting a Consumer Group

You can delete a consumer group using either of the following methods:

- Method 1: Delete a consumer group on the console.
- Method 2: Use **Kafka CLI** to delete a consumer group. (Ensure that the Kafka instance version is the same as the CLI version.)

Prerequisites

The status of the consumer group to be deleted is **EMPTY**.

Method 1: Deleting a Consumer Group on the Console

Step 1 Log in to the management console.

Step 2 Click 📍 in the upper left corner to select a region.

🕮 **NOTE**

Select the region where your Kafka instance is located.

Step 3 Click ☰ and choose **Application** > **Distributed Message Service (for Kafka)** to open the console of DMS for Kafka.

Step 4 Click the desired Kafka instance to view the instance details.

Step 5 In the navigation pane, choose the **Consumer Groups** tab.

Step 6 Delete consumer groups using either of the following methods:

- Select one or more consumer groups and click **Delete Consumer Group** above the consumer group list.
- In the row containing the consumer group you want to delete, click **Delete**.

NOTICE

A consumer group can be deleted only when its status is **EMPTY**.

Consumer group statuses include:

- **DEAD**: The consumer group has no member or metadata.
- **EMPTY**: The consumer group has metadata but has no member.
- **PREPARING_REBALANCE**: The consumer group is to be rebalanced.
- **COMPLETING_REBALANCE**: All members have joined the consumer group.
- **STABLE**: Members in the consumer group can consume messages normally.

Step 7 In the displayed **Delete Consumer Group** dialog box, click **Yes**.

----End

Method 2: Using the CLI to Delete a Consumer Group

The following uses Linux as an example.

Step 1 Download Kafka CLI **v1.1.0**, **v2.3.0**, or **v2.7.2**. Ensure that the Kafka instance and the CLI are of the same version.

Step 2 Use the CLI to connect to the Kafka instance. For details, see **Accessing a Kafka Instance Without SASL** or **Accessing a Kafka Instance with SASL**.

Step 3 In the ***/{directory where the CLI is located}*/kafka_*{version}*/bin/** directory, run the following command to delete a consumer group:

./kafka-consumer-groups.sh --bootstrap-server *{Kafka instance connection address}* **--delete --group** *{consumer group name}*

```
[root@zk-server-1 bin]# ./kafka-consumer-groups.sh --bootstrap-server 
192.168.1.245:9091,192.168.1.86:9091,192.168.1.128:9091 --delete --group bbbb
Note: This will not show information about old Zookeeper-based consumers.
Deletion of requested consumer groups ('bbbb') was successful.
```

NOTE

If SASL authentication is enabled for the Kafka instance, the **--command-config** ***{consumer.properties file with SASL authentication}*** parameter must be added to the preceding commands. For details about the **consumer.properties** file, see **Accessing a Kafka Instance with SASL**.

----End

9.3 Resetting the Consumer Offset

Resetting the consumer offset is to change the retrieval position of a consumer.

NOTICE

Messages may be retrieved more than once after the offset is reset. Exercise caution when performing this operation.

Prerequisites

The consumer offset cannot be reset on the fly. You must first stop retrieval of the desired consumer group.

NOTICE

After a client is stopped, the server considers the client offline only after the time period specified in **ConsumerConfig.SESSION_TIMEOUT_MS_CONFIG** (1000 ms by default).

Procedure

Step 1 Log in to the management console.

Step 2 Click 📍 in the upper left corner to select a region.

NOTE

Select the region where your Kafka instance is located.

Step 3 Click ☰ and choose **Application** > **Distributed Message Service (for Kafka)** to open the console of DMS for Kafka.

Step 4 Click the desired Kafka instance to view the instance details.

Step 5 In the navigation pane, choose the **Consumer Groups** tab.

Step 6 Click the name of the desired consumer group.

Step 7 On the **Consumer Offset** tab page, you can perform the following operations:

- To reset the consumer offset of all partitions of a single topic, click **Reset Consumer Offset** in the row containing the desired topic.
- To reset the consumer offset of a single partition of a single topic, click **Reset Consumer Offset** in the row containing the desired partition.

Step 8 In the displayed **Reset Consumer Offset** dialog box, set the parameters by referring to **Table 9-1**.

Table 9-1 Parameters for resetting the consumer offset

Parameter	Description
Reset By	You can reset an offset by: ● Time: Reset the offset to the specified time. ● Offset: Reset the offset to the specified position.
Time	Set this parameter if **Reset By** is set to **Time**. Select a time point. After the reset is complete, retrieval starts from this time point. ● **Earliest**: earliest offset ● **Custom Time Range**: a custom time point ● **Latest**: latest offset
Offset	Set this parameter if **Reset By** is set to **Offset**. Enter an offset, which is greater than or equal to 0. After the reset is complete, retrieval starts from this offset.

Step 9 Click **OK**.

Step 10 Click **Yes** in the confirmation dialog box. The consumer offset is reset.

----End

9.4 Viewing Consumer Connection Addresses

You can view connection addresses of consumers using either of the following methods:

- Method 1: View consumer connection addresses on the management console.
- Method 2: View consumer connection addresses on Kafka Manager.

NOTE

- The connection address of a consumer can be viewed only when the consumer is connected to a Kafka instance.
- Due to cache reasons, the consumer connection addresses displayed on Kafka Manager may not be used currently. To solve this problem, restart Kafka Manager.

Method 1: Viewing Consumer Addresses on Console

Step 1 Log in to the management console.

Step 2 Click 📍 in the upper left corner to select a region.

NOTE

Select the region where your Kafka instance is located.

Step 3 Click ☰ and choose **Application** > **Distributed Message Service (for Kafka)** to open the console of DMS for Kafka.

Step 4 Click the desired Kafka instance to view the instance details.

Step 5 In the navigation pane, choose **Consumer Groups**.

Step 6 Click the desired consumer group.

Step 7 On the **Consumers** tab page, view the consumer addresses.

----End

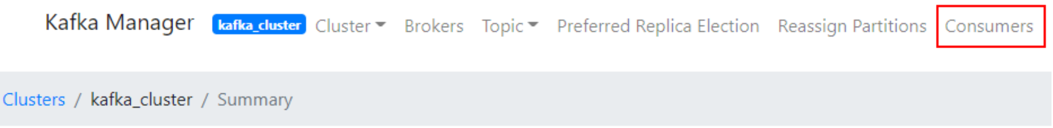
Method 2: Viewing Consumer Addresses on Kafka Manager

Step 1 **Log in to Kafka Manager**.

Step 2 Click **kafka_cluster** to go to the cluster details page.

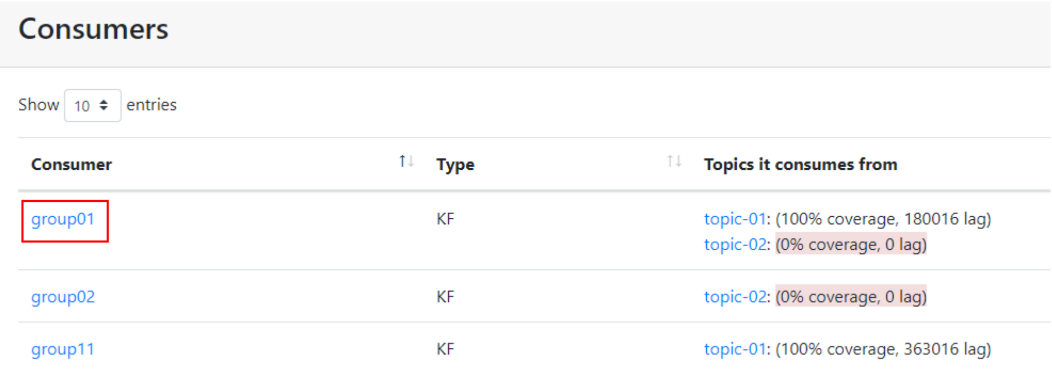
Step 3 On the top menu bar, choose **Consumers**.

Figure 9-3 Navigation bar



Step 4 Click the desired consumer group to view the topics that the group has subscribed to.

Figure 9-4 Consumer group list



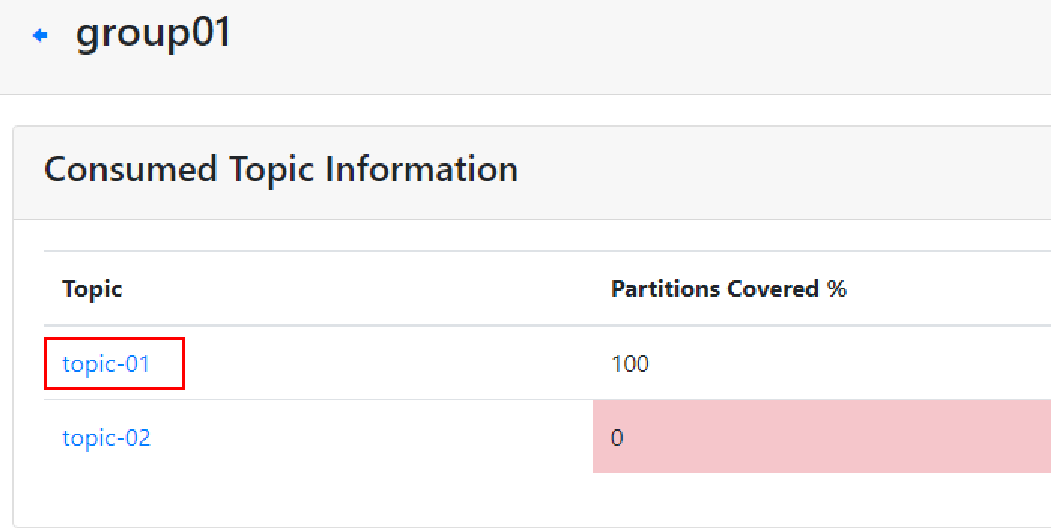
Consumers

Show 10 entries

Consumer	Type	Topics it consumes from
group01	KF	topic-01: (100% coverage, 180016 lag) topic-02: (0% coverage, 0 lag)
group02	KF	topic-02: (0% coverage, 0 lag)
group11	KF	topic-01: (100% coverage, 363016 lag)

Step 5 Click the desired topic to go to the topic details page.

Figure 9-5 Topics that the consumer group has subscribed to



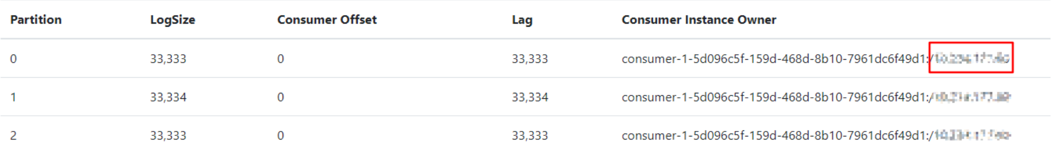
group01

Consumed Topic Information

Topic	Partitions Covered %
topic-01	100
topic-02	0

Step 6 In the **Consumer Instance Owner** column, view the consumer connection address.

Figure 9-6 Topic details page



Partition	LogSize	Consumer Offset	Lag	Consumer Instance Owner
0	33,333	0	33,333	consumer-1-5d096c5f-159d-468d-8b10-7961dc6f49d1:/[redacted]
1	33,334	0	33,334	consumer-1-5d096c5f-159d-468d-8b10-7961dc6f49d1:/[redacted]
2	33,333	0	33,333	consumer-1-5d096c5f-159d-468d-8b10-7961dc6f49d1:/[redacted]

----End

10 Managing Kafka Quotas

10.1 Creating a Quota

Scenario

On the console, you can control the message production and consumption rate limits for users or clients.

Operation Impact

- When the quota is reached, production/consumption latency increases.
- If the quota is small and the production rate is high, production may time out and messages may be lost. As a result, some messages fail to be produced.
- If the initial production/consumption traffic is heavy, and a small quota is set, the production/consumption latency increases and some messages fail to be produced. To ensure stable production and consumption, you are advised to first set the quota to half the traffic, and then half the quota each time you set it until the target quota is reached. For example, if the initial production traffic is 100 MB/s, you can set the production limit to 50 MB/s first. After production becomes stable, change the production limit to 25 MB/s until the target limit is reached.

Prerequisites

- To control user traffic, enable SASL_SSL when creating a Kafka instance and then obtain the username on the **Users** page on the console.
- To control client traffic, obtain the client ID from the client configuration.

Creating a User or Client Quota

Step 1 Log in to the management console.

Step 2 Click 📍 in the upper left corner to select a region.

NOTE

Select the region where your Kafka instance is located.

Step 3 Click ☰ and choose **Application** > **Distributed Message Service (for Kafka)** to open the console of DMS for Kafka.

Step 4 Click the desired Kafka instance to view the instance details.

Step 5 In the navigation pane, choose **Kafka Quotas**.

Step 6 In the upper left corner, click **Create Quota**. The **Create Quota** slide panel is displayed.

Step 7 Set quota parameters.

Table 10-1 Quota parameters

Parameter	Description
Username	Enter the name of the user to apply the quota to. To apply the quota to all users, click **Use Default** next to **Username**. After the quota is created, the username cannot be changed.
Client ID	Enter the ID of the client to which the quota applies. To apply the quota to all clients, click **Use Default** next to **Client ID**. After the quota is created, the client ID cannot be changed.
Production Limit	Set an upper limit on the production rate. The unit is byte/s. If this parameter is left blank, no limit is set.
Consumption Limit	Set an upper limit on the consumption rate. The unit is byte/s. If this parameter is left blank, no limit is set.

NOTE

- If SASL is not enabled for the instance, **Username** is not displayed in the **Create Quota** slide panel.
- **Username** and **Client ID** cannot be both empty.
- **Production Limit** and **Consumption Limit** cannot be both empty.

Step 8 Click **OK**. The **Background Tasks** page is displayed. If the status of the quota creation task is **Successful**, the quota has been created.

Go to the **Kafka Quotas** page, and click **user**, **client**, **user-client**, or ▾ to view the created quota.

- **user**: View quotas that apply only to users.
- **client**: View quotas that apply only to clients.
- **user-client**: View quotas that apply to both users and clients.

----End

10.2 Modifying a Quota

Scenario

After creating quotas, you can change the production or consumption rate limits.

Procedure

Step 1 Log in to the management console.

Step 2 Click 📍 in the upper left corner to select a region.

NOTE

Select the region where your Kafka instance is located.

Step 3 Click ☰ and choose **Application** > **Distributed Message Service (for Kafka)** to open the console of DMS for Kafka.

Step 4 Click the desired Kafka instance to view the instance details.

Step 5 In the navigation pane, choose **Kafka Quotas**.

Step 6 In the row containing the quota to be edited, click **Edit**.

Step 7 Change the production limit or consumption limit, and click **OK**. The **Background Tasks** page is displayed. If the status of the quota modification task is **Successful**, the quota has been modified.

Go to the **Kafka Quotas** page and view the new production or consumption rate limit.

NOTE

Production Limit and **Consumption Limit** cannot be both empty.

----End

10.3 Deleting a Quota

Scenario

Delete a quota when it is no longer needed.

Procedure

Step 1 Log in to the management console.

Step 2 Click 📍 in the upper left corner to select a region.

NOTE

Select the region where your Kafka instance is located.

Step 3 Click ☰ and choose **Application** > **Distributed Message Service (for Kafka)** to open the console of DMS for Kafka.

Step 4 Click the desired Kafka instance to view the instance details.

Step 5 In the navigation pane, choose **Kafka Quotas**.

Step 6 In the row containing the quota to be deleted, click **Delete**.

Step 7 Click **Yes**. The **Background Tasks** page is displayed. If the status of the quota deletion task is **Successful**, the quota has been deleted.

----End

11 Modifying Kafka Parameters

Scenario

Your Kafka instances, topics, and consumers come with default configuration parameter settings. You can modify common parameters on the Kafka console. For details about parameters that are not listed on the console, see the **Kafka official website**.

Parameters of v1.1.0 instances are all static parameters. v2.3.0/v2.7 instances have both dynamic and static parameters.

- Dynamic parameters: Modifying dynamic parameters will not restart the instance.
- Static parameters: After static parameters are modified, you must manually restart the instance.

> **NOTE**
>
> Configuration parameters of some old instances cannot be modified. Check whether your instance parameters can be modified on the console. If they cannot be modified, contact customer service.

Prerequisites

You can modify configuration parameters of a Kafka instance when the instance is in the **Running** state.

Procedure

Step 1 Log in to the management console.

Step 2 Click 📍 in the upper left corner to select a region.

> **NOTE**
>
> Select the region where your Kafka instance is located.

Step 3 Click ☰ and choose **Application** > **Distributed Message Service (for Kafka)** to open the console of DMS for Kafka.

Step 4 Click the desired Kafka instance to view the instance details.

Step 5 On the **Parameters** tab page, click **Edit** in the row containing the parameter to modify. Parameters of v1.1.0 instances are described in **Table 11-1**. Parameters of v2.3.0/v2.7 instances are described in **Table 11-2** and **Table 11-3**.

Table 11-1 Static parameters (v1.1.0 instances)

Parameter	Description	Value Range	Default Value
min.insync.replicas	If a producer sets the acks parameter to **all** (or **-1**), the **min.insync.replicas** parameter specifies the minimum number of replicas that must acknowledge a write for the write to be considered successful.	1–3	1
message.max.bytes	Maximum length of a single message, in bytes.	0–10,485,760	10,485,760
unclean.leader.election.enable	Indicates whether to allow replicas not in the ISR set to be elected as the leader as a last resort, even though doing so may result in data loss.	**true** or **false**	true
connections.max.idle.ms	Idle connection timeout (in ms). Connections that are idle for the duration specified by this parameter will be closed.	5000–600,000	600,000
log.retention.hours	Duration (in hours) for retaining a log file. This parameter takes effect only for topics that have no aging time configured. If there is aging time configured for topics, it overrides this parameter.	1–168	72
max.connections.per.ip	The maximum number of connections allowed from each IP address. Request for new connections will be rejected once the limit is reached.	100–20,000	1000
group.max.session.timeout.ms	The maximum session timeout (in ms) for consumers. A longer timeout gives consumers more time to process messages between heartbeats but results in a longer time to detect failures.	6000–1,800,000	1,800,000

Parameter	Description	Value Range	Default Value
default.replication.factor	The default number of replicas configured for an automatically created topic.	1–3	3
num.partitions	The default number of partitions configured for each automatically created topic.	1–100	3
group.min.session.timeout.ms	The minimum session timeout (in ms) for consumers. A shorter timeout enables quicker failure detection but results in more frequent consumer heartbeating, which can overwhelm broker resources.	6000–300,000	6000

Table 11-2 Dynamic parameters (v2.3.0/v2.7 instances)

Parameter	Description	Value Range	Default Value
min.insync.replicas	If a producer sets the acks parameter to **all** (or **-1**), the **min.insync.replicas** parameter specifies the minimum number of replicas that must acknowledge a write for the write to be considered successful.	1–3	1
message.max.bytes	Maximum length of a single message, in bytes.	0–10,485,760	10,485,760
max.connections.per.ip	The maximum number of connections allowed from each IP address. Request for new connections will be rejected once the limit is reached.	100–20,000	1000
unclean.leader.election.enable	Indicates whether to allow replicas not in the ISR set to be elected as the leader as a last resort, even though doing so may result in data loss.	**true** or **false**	true

Table 11-3 Static parameters (v2.3.0/v2.7 instances)

Parameter	Description	Value Range	Default Value
connections.max.idle.ms	Idle connection timeout (in ms). Connections that are idle for the duration specified by this parameter will be closed.	5000–600,000	600,000
log.retention.hours	Duration (in hours) for retaining a log file. This parameter takes effect only for topics that have no aging time configured. If there is aging time configured for topics, it overrides this parameter.	1–168	72
group.max.session.timeout.ms	The maximum session timeout (in ms) for consumers. A longer timeout gives consumers more time to process messages between heartbeats but results in a longer time to detect failures.	6000–1,800,000	1,800,000
default.replication.factor	The default number of replicas configured for an automatically created topic.	1–3	3
num.partitions	The default number of partitions configured for each automatically created topic.	1–100	3
group.min.session.timeout.ms	The minimum session timeout (in ms) for consumers. A shorter timeout enables quicker failure detection but results in more frequent consumer heartbeating, which can overwhelm broker resources.	6000–300,000	6000

NOTE

- To modify multiple dynamic or static parameters at a time, click **Modify** above the parameter list.
- If you want to restore the default values, click **Restore Default** in the row containing the desired parameter.

Step 6 Click **Save**.

NOTE

Modifying dynamic parameters will not restart the instance. **Static parameter modification requires manual restart of the instance.**

----End

12 Quotas

What Is a Quota?

A quota is a limit on the quantity or capacity of a certain type of service resources that you can use, for example, the maximum number of Kafka instances that you can create.

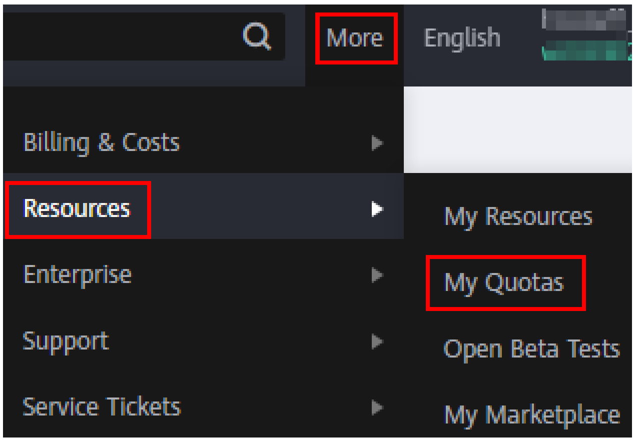
If a quota cannot meet your needs, apply for a higher quota.

How Do I View My Quota?

1. Log in to the management console.
2. In the upper right corner of the page, choose **Resources** > **My Quotas**.

   The **Quotas** page is displayed.

   Figure 12-1 My Quotas

3. On the **Quotas** page, view the used and total quotas of resources.

   If a quota cannot meet your needs, apply for a higher quota by performing the following operations.

How Do I Increase My Quota?

1. Log in to the management console.
2. In the upper right corner of the page, choose **Resources** > **My Quotas**.

The **Service Quota** page is displayed.

3. Click **Increase Quota**.
4. On the **Create Service Ticket** page, set the parameters.

   In the **Problem Description** area, enter the required quota and the reason for the quota adjustment.
5. Read the agreements and confirm that you agree to them, and then click **Submit**.

13 Monitoring

13.1 Viewing Metrics

Scenario

Cloud Eye monitors Kafka instance metrics in real time. You can view these metrics on the Cloud Eye console.

Prerequisites

At least one Kafka instance has been created. The instance has at least one available message.

Procedure

Step 1 Log in to the management console.

Step 2 Click 📍 in the upper left corner to select a region.

NOTE

Select the region where your Kafka instance is located.

Step 3 Click ☰ and choose **Application** > **Distributed Message Service (for Kafka)** to open the console of DMS for Kafka.

Step 4 View the instance metrics using either of the following methods:

- Click 🖥 next to a Kafka instance name. On the Cloud Eye console, view the metrics of the instance, brokers, topics, and consumer groups. Metric data is reported to Cloud Eye every minute.
- Click the desired Kafka instance to view its details. In the navigation pane, choose **Monitoring** view. On the displayed page, view the metrics of the instance, brokers, topics, and consumer groups. Metric data is reported to Cloud Eye every minute.

----End

13.2 Kafka Metrics

Introduction

This section describes metrics reported by DMS for Kafka to Cloud Eye as well as their namespaces and dimensions. You can use the Cloud Eye console or **APIs** to query the Kafka metrics and alarms, or view Kafka instance metrics on the **Monitoring** page of the DMS for Kafka console.

For example, you can call the **API** to query the monitoring data of the **Disk Capacity Usage** metric.

Namespace

SYS.DMS

Instance Metrics

Table 13-1 Instance metrics

Metric ID	Metric Name	Description	Value Range	Monitored Object	Monitoring Period (Raw Data)
current_partitions	Partitions	Number of used partitions in the instance Unit: count	0–1800	Kafka instance	1 minute
current_topics	Topics	Number of created topics in the instance Unit: count	0–1800	Kafka instance	1 minute
group_msgs	Accumulated Messages	Total number of accumulated messages in all consumer groups of the instance Unit: count	0–1,000,000,000	Kafka instance	1 minute

Broker Metrics

Table 13-2 Broker metrics

Metric ID	Metric Name	Description	Value Range	Monitored Object	Monitoring Period (Raw Data)
broker_data_size	Message Size	Total size of messages in the broker Unit: byte, KB, MB, GB, TB or PB	0–5,000,000,000,000	Kafka instance broker	1 minute
broker_messages_in_rate	Message Creation Rate	Number of messages created per second Unit: count/s	0–500,000	Kafka instance broker	1 minute
broker_bytes_out_rate	Message Retrieval	Number of bytes retrieved per second Unit: byte/s, KB/s, MB/s, or GB/s	0–500,000,000	Kafka instance broker	1 minute
broker_bytes_in_rate	Message Creation	Number of bytes created per second Unit: byte/s, KB/s, MB/s, or GB/s	0–500,000,000	Kafka instance broker	1 minute
broker_public_bytes_in_rate	Public Inbound Traffic	Inbound traffic over public networks per second Unit: byte/s, KB/s, MB/s, or GB/s **NOTE** You can view this metric on the EIP console if public access has been enabled and EIPs have been assigned to the instance.	0–500,000,000	Kafka instance broker	1 minute
broker_public_bytes_out_rate	Public Outbound Traffic	Outbound traffic over public networks per second Unit: byte/s, KB/s, MB/s, or GB/s **NOTE** You can view this metric on the EIP console if public access has been enabled and EIPs have been assigned to the instance.	0–500,000,000	Kafka instance broker	1 minute

Metric ID	Metric Name	Description	Value Range	Monitored Object	Monitoring Period (Raw Data)
broker_fetch_mean	Average Message Retrieval Processing Duration	Average time that the broker spends processing message retrieval requests Unit: ms	0–10,000	Kafka instance broker	1 minute
broker_produce_mean	Average Message Creation Processing Duration	Average time that the broker spends processing message creation requests Unit: ms	0–10,000	Kafka instance broker	1 minute
broker_cpu_core_load	Average Load per CPU Core	Average load of each CPU core of the Kafka VM Unit: %	0–20	Kafka instance broker	1 minute
broker_disk_usage	Disk Capacity Usage	Disk usage of the Kafka VM Unit: %	0–100	Kafka instance broker	1 minute
broker_memory_usage	Memory Usage	Memory usage of the Kafka VM Unit: %	0–100	Kafka instance broker	1 minute

Metric ID	Metric Name	Description	Value Range	Monitored Object	Monitoring Period (Raw Data)
broker_heap_usage	JVM Heap Memory Usage of Kafka	Heap memory usage of the Kafka JVM Unit: %	0–100	Kafka instance broker	1 minute
broker_alive	Broker Alive	Whether the Kafka broker is alive	● **1**: alive ● **0**: not alive	Kafka instance broker	1 minute
broker_connections	Connections	Total number of TCP connections on the Kafka broker Unit: count	0–65,535	Kafka instance broker	1 minute
broker_cpu_usage	CPU Usage	CPU usage of the Kafka VM Unit: %	0–100	Kafka instance broker	1 minute
broker_disk_read_await	Average Disk Read Time	Average time for each disk I/O read in the monitoring period Unit: ms	> 0	Kafka instance broker	1 minute
broker_disk_write_await	Average Disk Write Time	Average time for each disk I/O write in the monitoring period Unit: ms	> 0	Kafka instance broker	1 minute
broker_total_bytes_in_rate	Inbound Traffic	Inbound traffic per second Unit: byte/s	0–1,000,000,000	Kafka instance broker	1 minute
broker_total_bytes_out_rate	Outbound Traffic	Outbound traffic per second Unit: byte/s	0–1,000,000,000	Kafka instance broker	1 minute

Metric ID	Metric Name	Description	Value Range	Monitored Object	Monitoring Period (Raw Data)
broker_disk_read_rate	Disk Read Speed	Read traffic on the disk Unit: byte/s, KB/s, MB/s, or GB/s	≥ 0	Kafka instance broker	1 minute
broker_disk_write_rate	Disk Write Speed	Write traffic on the disk Unit: byte/s, KB/s, MB/s, or GB/s	≥ 0	Kafka instance broker	1 minute

Topic Metrics

Table 13-3 Topic metrics

Metric ID	Metric Name	Description	Value Range	Monitored Object	Monitoring Period (Raw Data)
topic_bytes_in_rate	Message Creation	Number of bytes created per second Unit: byte/s, KB/s, MB/s, or GB/s **NOTE** This metric is available only when **Scope** is set to **Basic monitoring** on the **By Topic** tab page.	0–500,000,000	Topic in a Kafka instance	1 minute

Metric ID	Metric Name	Description	Value Range	Monitored Object	Monitoring Period (Raw Data)
topic_bytes_out_rate	Message Retrieval	Number of bytes retrieved per second Unit: byte/s, KB/s, MB/s, or GB/s **NOTE** This metric is available only when **Scope** is set to **Basic monitoring** on the **By Topic** tab page.	0–500,000,000	Topic in a Kafka instance	1 minute
topic_data_size	Message Size	Total size of messages in the queue Unit: byte, KB, MB, GB, TB or PB **NOTE** This metric is available only when **Scope** is set to **Basic monitoring** on the **By Topic** tab page.	0–5,000,000,000,000	Topic in a Kafka instance	1 minute
topic_messages	Total Messages	Total number of messages in the queue Unit: count **NOTE** This metric is available only when **Scope** is set to **Basic monitoring** on the **By Topic** tab page.	≥ 0	Topic in a Kafka instance	1 minute
topic_messages_in_rate	Message Creation Rate	Number of messages created per second Unit: count/s **NOTE** This metric is available only when **Scope** is set to **Basic monitoring** on the **By Topic** tab page.	0–500,000	Topic in a Kafka instance	1 minute

Metric ID	Metric Name	Description	Value Range	Monitored Object	Monitoring Period (Raw Data)
partition_messages	Partition Messages	Total number of messages in the partition Unit: count **NOTE** This metric is available only when **Scope** is set to **Partition monitoring** on the **By Topic** tab page.	≥ 0	Topic in a Kafka instance	1 minute
produced_messages	Created Messages	Number of messages that have been created Unit: count **NOTE** This metric is available only when **Scope** is set to **Partition monitoring** on the **By Topic** tab page.	≥ 0	Topic in a Kafka instance	1 minute

Consumer Group Metrics

Table 13-4 Consumer group metrics

Metric ID	Metric Name	Description	Value Range	Monitored Object	Monitoring Period (Raw Data)
messages_consumed	Retrieved Messages	Number of messages that have been retrieved in the consumer group Unit: count **NOTE** This metric is available only when **Topic** is set to a specific topic name and **Monitoring Type** is set to **Partition monitoring** on the **By Consumer Group** tab page.	≥ 0	Consumer group of a Kafka instance	1 minute
messages_remained	Available Messages	Number of messages that can be retrieved in the consumer group Unit: count **NOTE** This metric is available only when **Topic** is set to a specific topic name and **Monitoring Type** is set to **Partition monitoring** on the **By Consumer Group** tab page.	≥ 0	Consumer group of a Kafka instance	1 minute
topic_messages_remained	Topic Available Messages	Number of remaining messages that can be retrieved from the specified topic in the consumer group Unit: Count **NOTE** This metric is available only when **Topic** is set to a specific topic name and **Monitoring Type** is set to **Basic monitoring** on the **By Consumer Group** tab page.	0 to 2^{63}–1	Consumer group of a Kafka instance	1 minute

Metric ID	Metric Name	Description	Value Range	Monitored Object	Monitoring Period (Raw Data)
topic_messages_consumed	Topic Retrieved Messages	Number of messages that have been retrieved from the specified topic in the consumer group Unit: Count **NOTE** This metric is available only when **Topic** is set to a specific topic name and **Monitoring Type** is set to **Basic monitoring** on the **By Consumer Group** tab page.	0 to 2^{63}–1	Consumer group of a Kafka instance	1 minute
consumer_messages_remained	Consumer Available Messages	Number of remaining messages that can be retrieved in the consumer group Unit: Count **NOTE** This metric is available only when **Topic** is set to **All topics** on the **By Consumer Group** tab page.	0 to 2^{63}–1	Consumer group of a Kafka instance	1 minute
consumer_messages_consumed	Consumer Retrieved Messages	Number of messages that have been retrieved in the consumer group Unit: Count **NOTE** This metric is available only when **Topic** is set to **All topics** on the **By Consumer Group** tab page.	0 to 2^{63}–1	Consumer group of a Kafka instance	1 minute

Dimension

Key	Value
kafka_instance_id	Kafka instance
kafka_broker	Kafka instance broker

Key	Value
kafka_topics	Kafka instance topic
kafka_partitions	Partition in a Kafka instance
kafka_groups-partitions	Partition consumer group in a Kafka instance
kafka_groups_topics	Topic consumer group in a Kafka instance
kafka_groups	Consumer group of a Kafka instance

13.3 Configuring Alarm Rules

This section describes the alarm rules of some metrics and how to configure them. In actual services, you are advised to configure alarm rules for metrics based on the following alarm policies:

Table 13-5 Kafka instance metrics to configure alarm rules for

Metric ID	Metric	Alarm Policy	Description	Handling Suggestion
broker_disk_usage	Disk Capacity Usage	Alarm threshold: original value > 80% Number of consecutive periods: 1 Alarm severity: critical	Disk usage of the Kafka VM	Modify the instance **storage space**. For details, see **Modifying Instance Specifications**.
broker_cpu_core_load	Average Load per CPU Core	Alarm threshold: original value > 2 Number of consecutive periods: 3 Alarm severity: major	Average load of each CPU core of the Kafka VM.	Check whether the metric has been approaching or exceeding the alarm threshold for a long time. If yes, modify the **number of brokers**. For details, see **Modifying Instance Specifications**.

Metric ID	Metric	Alarm Policy	Description	Handling Suggestion
broker_memory_usage	Memory Usage	Alarm threshold: original value > 90% Number of consecutive periods: 3 Alarm severity: critical	Memory usage of the Kafka VM.	Modify the **number of brokers**. For details, see **Modifying Instance Specifications**.
current_partitions	Partitions	Alarm threshold: original value > 90% of the maximum allowed number of partitions. The partition limit varies depending on instance specifications. For details, see **Specifications**. Number of consecutive periods: 1 Alarm severity: major	Number of used partitions in the instance.	If new topics are required, modify the number of brokers, or split the service to multiple instances. For details about how to modify the number of brokers, see **Modifying Instance Specifications**.
broker_cpu_usage	CPU Usage	Alarm threshold: original value > 90% Number of consecutive periods: 3 Alarm severity: major	CPU usage of the Kafka VM.	Check whether the metric has been approaching or exceeding the alarm threshold for a long time. If yes, modify the **number of brokers**. For details, see **Modifying Instance Specifications**.

Metric ID	Metric	Alarm Policy	Description	Handling Suggestion
group_msgs	Accumulated Messages	Alarm threshold: original value > 90% of the upper limit. The upper limit is customized. Number of consecutive periods: 1 Alarm severity: major	Total number of accumulated messages in all consumer groups of the instance	Delete idle consumer groups, if any. You can also accelerate message retrieval, for example, by increasing the number of consumers.
topic_messages_remained	Topic Available Messages	Alarm threshold: original value > 90% of the upper limit. The upper limit is customized. Number of consecutive periods: 1 Alarm severity: major	Number of remaining messages that can be retrieved from the specified topic in the consumer group.	Check whether the consumer code logic is correct, for example, by checking whether the consumer stops consuming messages due to an exception. You can also accelerate message retrieval, for example, by adding topic consumers. Ensure that the number of partitions is greater than or equal to the number of consumers.

Procedure

Step 1 Log in to the management console.

Step 2 Click 📍 in the upper left corner to select a region.

NOTE

Select the region where your Kafka instance is located.

Step 3 Click ☰ and choose **Application** > **Distributed Message Service (for Kafka)** to open the console of DMS for Kafka.

Step 4 Click 🖥 next to a Kafka instance name.

You are redirected to the Cloud Eye console page displaying metrics of the selected instance.

Step 5 Hover the mouse pointer over a metric and click + to create an alarm rule for the metric.

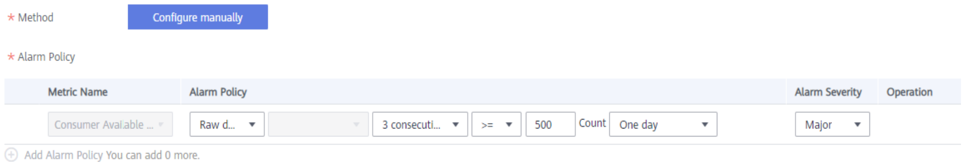
Step 6 Specify the alarm details.

For more information about creating alarm rules, see **Creating an Alarm Rule**.

1. Set the alarm name and description.
2. Specify the alarm policy and alarm severity.

   As shown in the following figure, if the original disk capacity usage exceeds 85% for three consecutive periods, an alarm is generated. If the alarm is not handled on time, an alarm notification is sent.

   Figure 13-1 Setting the alarm policy and alarm severity

   * Method: Configure manually

   * Alarm Policy

Metric Name	Alarm Policy							Alarm Severity	Operation
Consumer Available ...	Raw d...		3 consecuti...	>=	500	Count	One day	Major	

3. Set the alarm notification configurations. If you enable **Alarm Notification**, set the validity period, notification object, and trigger condition.
4. Click **Create**.

----End

14 Auditing

14.1 Operations Logged by CTS

With Cloud Trace Service (CTS), you can record operations associated with DMS for Kafka for later query, audit, and backtrack operations.

Table 14-1 DMS for Kafka operations that can be recorded by CTS

Operation	Resource Type	Trace Name
Successfully creating an order for creating an instance	kafka	createDMSInstanceOrderSuccess
Successfully creating an instance	kafka	createDMSInstanceTaskSuccess
Failing to create an order for creating an instance	kafka	createDMSInstanceOrderFailure
Failing to create an instance	kafka	createDMSInstanceTaskFailure
Successfully deleting an instance that failed to be created	kafka	deleteDMSCreateFailureInstancesSuccess
Failing to delete an instance that failed to be created	kafka	deleteDMSCreateFailureInstancesFailure
Successfully deleting an instance	kafka	deleteDMSInstanceTaskSuccess
Failing to delete an instance	kafka	deleteDMSInstanceTaskFailure

Operation	Resource Type	Trace Name
Deleting multiple instance tasks at a time	kafka	batchDeleteDMSInstanceTask
Successfully submitting a request to delete multiple instances at a time	kafka	batchDeleteDMSInstanceSuccess
Successfully deleting multiple instances at a time	kafka	batchDeleteDMSInstanceTaskSuccess
Failing to submit a request to delete multiple instances at a time	kafka	batchDeleteDMSInstanceFailure
Failing to delete multiple instances at a time	kafka	batchDeleteDMSInstanceTaskFailure
Successfully submitting a request to modify an instance order	kafka	modifyDMSInstanceOrderSuccess
Failing to submit a request to modify an instance order	kafka	modifyDMSInstanceOrderFailure
Successfully submitting a request to scale up an instance	kafka	extendDMSInstanceSuccess
Successfully scaling up an instance	kafka	extendDMSInstanceTaskSuccess
Failing to submit a request to scale up an instance	kafka	extendDMSInstanceFailure
Failing to scale up an instance	kafka	extendDMSInstanceTaskFailure
Successfully submitting a request to reset instance password	kafka	resetDMSInstancePasswordSuccess
Failing to submit a request to reset instance password	kafka	resetDMSInstancePasswordFailure

Operation	Resource Type	Trace Name
Successfully submitting a request to restart an instance	kafka	restartDMSInstanceSuccess
Successfully restarting an instance	kafka	restartDMSInstanceTaskSuccess
Failing to submit a request to restart an instance	kafka	restartDMSInstanceFailure
Failing to restart an instance	kafka	restartDMSInstanceTaskFailure
Successfully submitting a request to restart multiple instances at a time	kafka	batchRestartDMSInstanceSuccess
Successfully restarting multiple instances at a time	kafka	batchRestartDMSInstanceTaskSuccess
Failing to submit a request to restart multiple instances at a time	kafka	batchRestartDMSInstanceFailure
Failing to restart multiple instances at a time	kafka	batchRestartDMSInstanceTaskFailure
Successfully submitting a request to modify instance information	kafka	modifyDMSInstanceInfoSuccess
Successfully modifying instance information	kafka	modifyDMSInstanceInfoTaskSuccess
Failing to submit a request to modify instance information	kafka	modifyDMSInstanceInfoFailure
Failing to modify instance information	kafka	modifyDMSInstanceInfoTaskFailure
Successfully deleting a background task	kafka	deleteDMSBackendJobSuccess
Failing to delete a background task	kafka	deleteDMSBackendJobFailure
Successfully freezing an instance	kafka	freezeDMSInstanceTaskSuccess

Operation	Resource Type	Trace Name
Failing to freeze an instance	kafka	freezeDMSInstanceTaskFailure
Successfully unfreezing an instance	kafka	unfreezeDMSInstanceTaskSuccess
Failing to unfreeze an instance	kafka	unfreezeDMSInstanceTaskFailure
Successfully creating a topic for a Kafka instance	kafka	Kafka_create_topicSuccess
Failing to create a topic for a Kafka instance	kafka	Kafka_create_topicFailure
Successfully deleting a topic from a Kafka instance	kafka	Kafka_delete_topicsSuccess
Failing to delete a topic for a Kafka instance	kafka	Kafka_delete_topicsFailure
Successfully enabling automatic topic creation	kafka	enable_auto_topicSuccess
Failing to enable automatic topic creation	kafka	enable_auto_topicFailure
Successfully resetting the consumer offset	kafka	Kafka_reset_consumer_offsetSuccess
Failing to reset the consumer offset	kafka	Kafka_reset_consumer_offsetFailure
Successfully creating a user	kafka	createUserSuccess
Failing to create a user	kafka	createUserFailure
Successfully deleting a user	kafka	deleteUserSuccess
Failing to delete a user	kafka	deleteUserFailure
Successfully updating user policies	kafka	updateUserPoliciesTaskSuccess

Operation	Resource Type	Trace Name
Failing to update user policies	kafka	updateUserPoliciesTaskFailure

14.2 Viewing Audit Logs

See **Querying Real-Time Traces**.