Cloud Search Service

Getting Started

Issue 01

Date 2024-10-12

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.

Huawei Cloud Computing Technologies Co., Ltd.

Address: Huawei Cloud Data Center Jiaoxinggong Road
Qianzhong Avenue
Gui'an New District
Gui Zhou 550029
People's Republic of China

Website: https://www.huaweicloud.com/intl/en-us/

Contents

1 Using Elasticsearch for Data Search

This section provides an example of how an e-commerce website uses an Elasticsearch cluster to implement a product search function, including creating indexes, importing data, and searching for data.

Scenario Description

A women's clothing brand runs an e-commerce website. It has been using traditional databases to power a product search function for customers. However, as the website traffic increases, these traditional databases are struggling to keep up, leading to slow responses and low search accuracy. To improve shopping experience for customers, the e-commerce website plans to use Cloud Search Service (CSS) to provide the product search function.

Assume that the e-commerce website has the following data:

```
{
"products":[
{"productName":"Latest elegant shirts in autumn 2017","size":"L"}
{"productName":"Latest elegant shirts in autumn 2017","size":"M"}
{"productName":"Latest elegant shirts in autumn 2017","size":"S"}
{"productName":"Latest jeans in spring 2018","size":"M"}
{"productName":"Latest jeans in spring 2018","size":"S"}
{"productName":"Latest casual pants in spring 2017","size":"L"}
{"productName":"Latest casual pants in spring 2017","size":"S"}
]
}
```

Procedure

The following describes how to use an Elasticsearch cluster to implement a website search function.

Before starting to migrate data, make the necessary preparations. For details, see **Preparations**.

1. **Step 1: Creating a Cluster**: Create a non-security mode Elasticsearch cluster for data search.
2. **Step 2: Importing Data**: Use an open-source Elasticsearch API to import data on Kibana.
3. **Step 3: Searching for Data**: Perform full-text search and result aggregation and display on data in the Elasticsearch cluster.

4. **Step 4: Deleting Indexes**: Delete indexes that you no longer need to reclaim resources.

Preparations

You have registered with Huawei Cloud and performed real-name authentication. Make sure your account is not frozen or in arrears.

If you do not have a Huawei Cloud account, perform the following operations to create one:

1. Visit the **Huawei Cloud official website**.
2. In the upper right corner of the page, click **Register** and complete the registration as prompted.
3. Select the service agreement and click **Enable**.
4. Perform real-name authentication.
   - If your account is an individual account, see **Individual Real-Name Authentication**.
   - If your account is an enterprise account, see **Enterprise Real-Name Authentication**.

Step 1: Creating a Cluster

Create a non-security mode Elasticsearch cluster for data search.

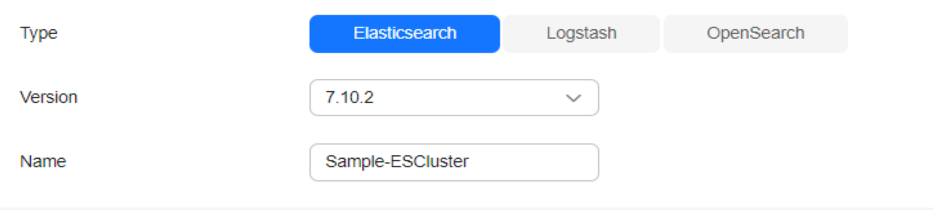
1. Log in to the CSS management console.
2. In the navigation pane on the left, choose **Clusters** > **Elasticsearch**.
3. Click **Create Cluster** in the upper right corner. The **Create Cluster** page is displayed.
4. Configure Billing Mode and AZ for the cluster.

Table 1-1 Billing mode and AZ parameters

Parameter	Description	Example Value
Billing Mode	Select **Yearly/Monthly** or **Pay-per-use**. ● **Yearly/monthly**: You pay for the cluster by year or month, in advance. The service duration ranges from one month to three years. If you plan to use a cluster for more than nine months, you are advised to purchase a yearly package for a better price. A yearly package costs the same as a 10 monthly package. ● **Pay-per-use**: You are billed by actual duration of use, with a billing cycle of one hour. For example, 58 minutes of usage will be rounded up to an hour and billed.	Pay-per-use
Region	Select the region where the cluster is located. ECSs in different regions cannot communicate with each other over an intranet. For lower network latency and quicker resource access, select the nearest region.	Hong Kong, China
AZ	Select AZs associated with the cluster region. A maximum of three AZs can be configured.	AZ 1

5. Configure basic cluster information.

Figure 1-1 Configuring cluster information



Type: Elasticsearch | Logstash | OpenSearch

Version: 7.10.2

Name: Sample-ESCluster

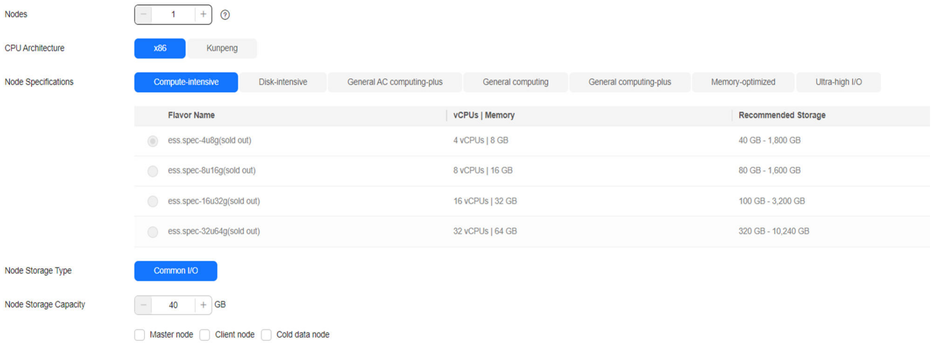
Table 1-2 Basic configuration parameters

Parameter	Description	Example Value
Cluster Type	Select **Elasticsearch**.	Elasticsearch

Parameter	Description	Example Value
Version	Select a cluster version from the drop-down list box.	7.10.2
Name	Cluster name, which contains 4 to 32 characters. Only letters, numbers, hyphens (-), and underscores (_) are allowed and the value must start with a letter.	Sample-ESCluster

6. Configure the cluster's node specifications

Figure 1-2 Configuring the cluster's node specifications



Nodes: 1

CPU Architecture: x86

Node Specifications: Compute-intensive

	Flavor Name	vCPUs | Memory	Recommended Storage
●	ess.spec-4u8g(sold out)	4 vCPUs | 8 GB	40 GB - 1,800 GB
○	ess.spec-8u16g(sold out)	8 vCPUs | 16 GB	80 GB - 1,600 GB
○	ess.spec-16u32g(sold out)	16 vCPUs | 32 GB	100 GB - 3,200 GB
○	ess.spec-32u64g(sold out)	32 vCPUs | 64 GB	320 GB - 10,240 GB

Node Storage Type: Common I/O

Node Storage Capacity: 40 GB

□ Master node □ Client node □ Cold data node

Table 1-3 Specification parameters

Parameter	Description	Example Value
Nodes	Set the number of nodes in the cluster. Select a number from 1 to 32.	1
CPU Architecture	The CPU architectures actually supported vary depending on the regional environment.	x86
Node Specifications	Select the specifications of cluster nodes.	ess.spec-4u8g
Node Storage Type	Select the storage type of cluster nodes.	High I/O
Node Storage Capacity	Node storage capacity. Its value range varies with node specifications. The node storage capacity must be a multiple of 20.	40GB
Master node	The Master node manages all node tasks in the cluster.	Unselect it.

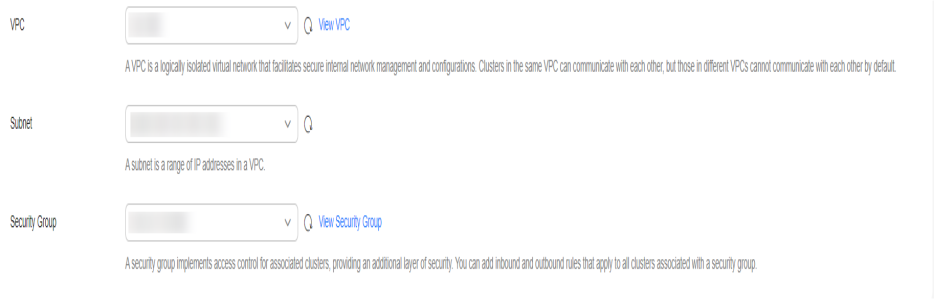
Parameter	Description	Example Value
Client node	Client nodes receive and coordinate external requests, such as search and write requests.	Unselect it.
Cold data node	Cold data nodes are used to store data that is not particularly sensitive to query latency in large quantities.	Unselect it.

7. Set the enterprise project.

   When creating a CSS cluster, you can bind an enterprise project to the cluster if you have enabled the enterprise project function. In this example, **default**, the default enterprise project, is selected.

8. Click **Next: Network**. Configure the cluster network.

Figure 1-3 Configuring networking



VPC

A VPC is a logically isolated virtual network that facilitates secure internal network management and configurations. Clusters in the same VPC can communicate with each other, but those in different VPCs cannot communicate with each other by default.

Subnet

A subnet is a range of IP addresses in a VPC.

Security Group

A security group implements access control for associated clusters, providing an additional layer of security. You can add inbound and outbound rules that apply to all clusters associated with a security group.

Table 1-4 Network configuration parameters

Parameter	Description	Example Value
VPC	Specify a VPC to isolate the cluster's network. **NOTE** The VPC must contain CIDRs. Otherwise, cluster creation will fail. By default, a VPC will contain CIDRs.	vpc-default
Subnet	A subnet provides dedicated network resources that are isolated from other networks, improving network security.	subnet-default
Security Group	A security group serves as a virtual firewall that provides access control policies for clusters. **NOTE** For enable cluster access, ensure that port 9200 is allowed by the security group.	default

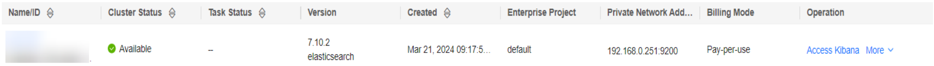
Parameter	Description	Example Value
Security Mode	After the security mode is enabled, communication will be encrypted and authentication required for the cluster.	Disable

9. Click **Next: Advanced Settings**. Configure automatic snapshot creation and other functions.

   This cluster is used only for getting started. Cluster snapshots and advanced functions are not required.

10. Click **Next: Confirm**. Check the configuration and click **Next** to create a cluster.
11. Click **Back to Cluster List** to switch to the **Clusters** page. The cluster you created is listed on the displayed page and its status is **Creating**. If the cluster is successfully created, its status will change to **Available**.

Figure 1-4 Creating a cluster



Name/ID	Cluster Status	Task Status	Version	Created	Enterprise Project	Private Network Add...	Billing Mode	Operation
[redacted]	Available	--	7.10.2 elasticsearch	Mar 21, 2024 09:17:5...	default	192.168.0.251:9200	Pay-per-use	Access Kibana More

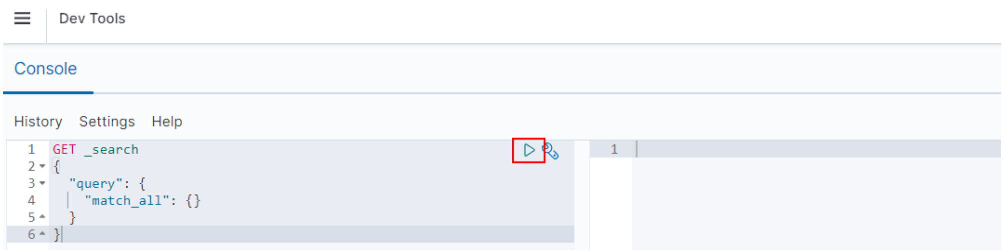
Step 2: Importing Data

There are several ways to import data to an Elasticsearch cluster. In this example, we use an open-source Elasticsearch API to import data on Kibana.

1. On the Elasticsearch cluster management page, select the created **Sample-ESCluster** cluster and click **Access Kibana** in the **Operation** column to access the Kibana console.
2. In the Kibana navigation pane on the left, choose **Dev Tools**.

   The text box on the left is the input box. The triangle icon in the upper right corner of the input box is the command execution button. The text box on the right area is the result output box.

Figure 1-5 Console page



```
GET _search
{
  "query": {
    "match_all": {}
  }
}
```

3. On the **Console** page, run the following command to create an index named **my_store**:

```
PUT /my_store
{
  "settings": {
    "number_of_shards": 1
  },
  "mappings": {
    "properties": {
      "productName": {
        "type": "text",
        "analyzer": "ik_smart"
        },
        "size": {
          "type": "keyword"
        }
      }
    }
  }
```

The command output is similar to the following:

```
{
  "acknowledged" : true,
  "shards_acknowledged" : true,
  "index" : "my_store"
}
```

4. On the **Console** page, run the following command to import data to the index named **my_store**:

```
POST /my_store/_doc/_bulk
{"index":{}}
{"productName":"Latest elegant shirts in autumn 2017","size":"L"}
{"index":{}}
{"productName":"Latest elegant shirts in autumn 2017","size":"M"}
{"index":{}}
{"productName":"Latest elegant shirts in autumn 2017","size":"S"}
{"index":{}}
{"productName":"Latest jeans in spring 2018","size":"M"}
{"index":{}}
{"productName":"Latest jeans in spring 2018","size":"S"}
{"index":{}}
{"productName":"Latest casual pants in spring 2017","size":"L"}
{"index":{}}
{"productName":"Latest casual pants in spring 2017","size":"S"}
```

If the value of the **errors** field in the command output is **false**, the data is imported successfully.

Step 3: Searching for Data

Perform full-text search and result aggregation and display on data in the Elasticsearch cluster.

- **Full-text search**

  If you access the e-commerce website and want to search for items whose names include "spring jeans", enter "spring jeans" to begin your search.

  Run the following command on Kibana:

```
GET /my_store/_search
{
  "query": {"match": {
    "productName": "spring jeans"
  }}
}
```

  The command output is similar to the following:

```
{
  "took" : 3,
  "timed_out" : false,
  "_shards" : {
    "total" : 1,
    "successful" : 1,
    "skipped" : 0,
    "failed" : 0
  },
  "hits" : {
    "total" : {
      "value" : 4,
      "relation" : "eq"
    },
    "max_score" : 1.7965372,
    "hits" : [
      {
        "_index" : "my_store",
        "_type" : "_doc",
        "_id" : "9xf6VHIBfClt6SDjw7H5",
        "_score" : 1.7965372,
        "_source" : {
          "productName": "Latest jeans in spring 2018",
          "size" : "M"
        }
      },
      {
        "_index" : "my_store",
        "_type" : "_doc",
        "_id" : "-Bf6VHIBfClt6SDjw7H5",
        "_score" : 1.7965372,
        "_source" : {
          "productName": "Latest jeans in spring 2018",
          "size" : "S"
        }
      },
      {
        "_index" : "my_store",
        "_type" : "_doc",
        "_id" : "-Rf6VHIBfClt6SDjw7H5",
        "_score" : 0.5945667,
        "_source" : {
          "productName": "Latest casual pants in spring 2017",
          "size" : "L"
        }
      },
      {
        "_index" : "my_store",
        "_type" : "_doc",
        "_id" : "-hf6VHIBfClt6SDjw7H5",
        "_score" : 0.5945667,
        "_source" : {
          "productName": "Latest casual pants in spring 2017",
          "size" : "S"
        }
      }
    ]
  }
}
```

– Elasticsearch supports IK word segmentation. The preceding command segments "spring jeans" into "spring" and "jeans".

– Elasticsearch supports full-text search. The preceding command searches for the information about all items whose names include "spring" or "jeans".

– Unlike traditional databases, Elasticsearch can return results in milliseconds by using inverted indexes.

– Elasticsearch supports sorting by score. In the command output, information about the first two items contains both "spring" and "jeans", while that about the last two items contain only "spring". Therefore, the first two items rank higher than the last two due to high keyword match.

- **Aggregated result display**

The e-commerce website displays aggregated results. For example, it classifies items corresponding to "spring" based on sizes so that you can count the number of items of different sizes.

Run the following result aggregation command on Kibana:

```
GET /my_store/_search
{
"query": {
"match": { "productName": "spring" }
},
"size": 0,
"aggs": {
"sizes": {
"terms": { "field": "size" }
}
}
}
```

The command output is similar to the following:

```
{
  "took" : 3,
  "timed_out" : false,
  "_shards" : {
    "total" : 1,
    "successful" : 1,
    "skipped" : 0,
    "failed" : 0
  },
  "hits" : {
    "total" : {
      "value" : 4,
      "relation" : "eq"
    },
    "max_score" : null,
    "hits" : [ ]
  },
  "aggregations" : {
    "sizes" : {
      "doc_count_error_upper_bound" : 0,
      "sum_other_doc_count" : 0,
      "buckets" : [
        {
          "key" : "S",
          "doc_count" : 2
        },
        {
          "key" : "L",
          "doc_count" : 1
        },
        {
          "key" : "M",
          "doc_count" : 1
        }
      ]
    }
  }
}
```

Step 4: Deleting Indexes

If an index is no longer used, run the following command on Kibana to delete the index to reclaim resources:

```
DELETE /my_store
```

The command output is similar to the following:

```
{
  "acknowledged" : true
}
```

Follow-up Operations

You can delete the cluster if you no longer need it.

📖 **NOTE**

After you delete a cluster, its data cannot be restored. Exercise caution when deleting a cluster.

1. Log in to the CSS management console.
2. In the navigation pane on the left, choose **Clusters** > **Elasticsearch**.
3. In the cluster list, locate the **Sample-ESCluster** cluster, and choose **More** > **Delete** in the **Operation** column.
4. In the confirmation dialog box, type in **DELETE**, and click **OK**.

2 Using OpenSearch for Data Search

This section provides an example of how an e-commerce website uses an OpenSearch cluster to implement a product search function, including creating indexes, importing data, and searching for data.

Scenario Description

A women's clothing brand runs an e-commerce website. It has been using traditional databases to power a product search function for customers. However, as the website traffic increases, these traditional databases are struggling to keep up, leading to slow responses and low search accuracy. To improve shopping experience for customers, the e-commerce website plans to use Cloud Search Service (CSS) to provide the product search function.

Assume that the e-commerce website has the following data:

```
{
"products":[
{"productName":"Latest elegant shirts in autumn 2017","size":"L"}
{"productName":"Latest elegant shirts in autumn 2017","size":"M"}
{"productName":"Latest elegant shirts in autumn 2017","size":"S"}
{"productName":"Latest jeans in spring 2018","size":"M"}
{"productName":"Latest jeans in spring 2018","size":"S"}
{"productName":"Latest casual pants in spring 2017","size":"L"}
{"productName":"Latest casual pants in spring 2017","size":"S"}
]
}
```

Procedure

The following describes how to use an OpenSearch cluster to implement a website search function.

Before starting to migrate data, make the necessary preparations. For details, see **Preparations**.

1. **Step 1: Creating a Cluster**: Create a non-security mode OpenSearch cluster for data search.
2. **Step 2: Importing Data**: Use an open-source Elasticsearch API to import data on OpenSearch Dashboards.
3. **Step 3: Searching for Data**: Perform full-text search and result aggregation and display on data in the OpenSearch cluster.

4. **Step 4: Deleting Indexes**: Delete indexes that you no longer need to reclaim resources.

Preparations

You have registered with Huawei Cloud and performed real-name authentication. Make sure your account is not frozen or in arrears.

If you do not have a Huawei Cloud account, perform the following operations to create one:

1. Visit the **Huawei Cloud official website**.
2. In the upper right corner of the page, click **Register** and complete the registration as prompted.
3. Select the service agreement and click **Enable**.
4. Perform real-name authentication.
   - If your account is an individual account, see **Individual Real-Name Authentication**.
   - If your account is an enterprise account, see **Enterprise Real-Name Authentication**.

Step 1: Creating a Cluster

Create a non-security mode OpenSearch cluster for data search.

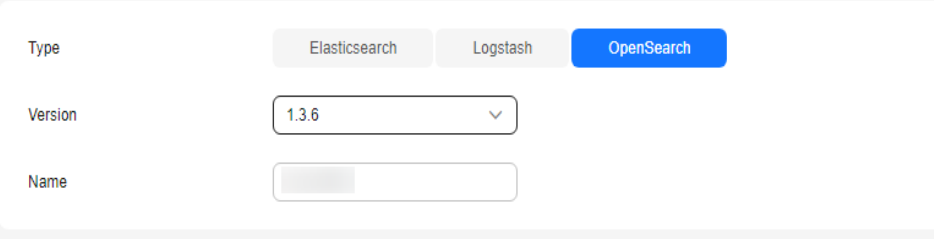
1. Log in to the CSS management console.
2. In the navigation pane on the left, choose **Clusters** > **OpenSearch**.
3. Click **Create Cluster** in the upper right corner. The **Create Cluster** page is displayed.
4. Configure Billing Mode and AZ for the cluster.

Table 2-1 Billing mode and AZ parameters

Parameter	Description	Example Value
Billing Mode	Select **Yearly/Monthly** or **Pay-per-use**. ● **Yearly/monthly**: You pay for the cluster by year or month, in advance. The service duration ranges from one month to three years. If you plan to use a cluster for more than nine months, you are advised to purchase a yearly package for a better price. A yearly package costs the same as a 10 monthly package. ● **Pay-per-use**: You are billed by actual duration of use, with a billing cycle of one hour. For example, 58 minutes of usage will be rounded up to an hour and billed.	Pay-per-use
Region	Select the region where the cluster is located. ECSs in different regions cannot communicate with each other over an intranet. For lower network latency and quicker resource access, select the nearest region.	Hong Kong, China
AZ	Select AZs associated with the cluster region. A maximum of three AZs can be configured.	AZ 1

5. Configure basic cluster information.

Figure 2-1 Configuring cluster information



Type: Elasticsearch | Logstash | OpenSearch

Version: 1.3.6

Name:

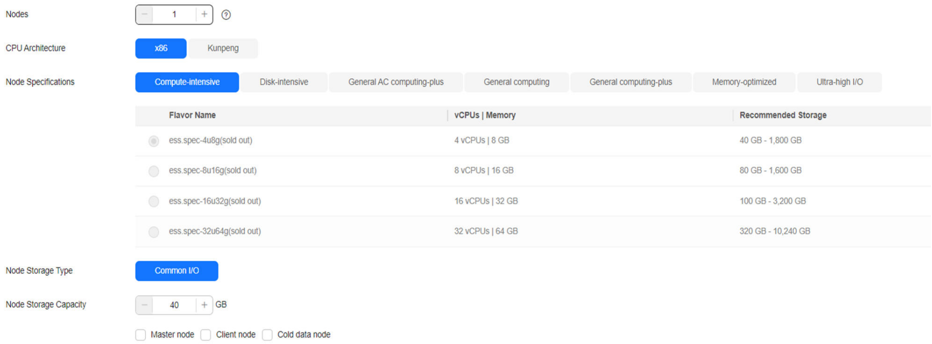
Table 2-2 Basic configuration parameters

Parameter	Description	Example Value
Cluster Type	Choose **OpenSearch**.	OpenSearch

Parameter	Description	Example Value
Version	Select a cluster version from the drop-down list box.	1.3.6
Name	Cluster name, which contains 4 to 32 characters. Only letters, numbers, hyphens (-), and underscores (_) are allowed and the value must start with a letter.	Sample-OSCluster

6. Configure the cluster's node specifications

Figure 2-2 Configuring the cluster's node specifications



Nodes: 1

CPU Architecture: x86

Node Specifications: Compute-intensive

Flavor Name	vCPUs | Memory	Recommended Storage
○ ess.spec-4u8g(sold out)	4 vCPUs | 8 GB	40 GB - 1,800 GB
○ ess.spec-8u16g(sold out)	8 vCPUs | 16 GB	80 GB - 1,600 GB
○ ess.spec-16u32g(sold out)	16 vCPUs | 32 GB	100 GB - 3,200 GB
○ ess.spec-32u64g(sold out)	32 vCPUs | 64 GB	320 GB - 10,240 GB

Node Storage Type: Common I/O

Node Storage Capacity: 40 GB

□ Master node □ Client node □ Cold data node

Table 2-3 Specification parameters

Parameter	Description	Example Value
Nodes	Number of nodes in a cluster. Select a number from 1 to 32.	1
CPU Architecture	Both **x86** and **Kunpeng** are supported. The supported type is determined by the actual regional environment.	x86
Node Specifications	Select the specifications of cluster nodes.	ess.spec-4u8g
Node Storage Type	Select the storage type of cluster nodes.	Common I/O
Node Storage Capacity	Node storage capacity. Its value range varies with node specifications. The node storage capacity must be a multiple of 20.	40GB
Master node	The Master node manages all node tasks in the cluster.	Unselect it.

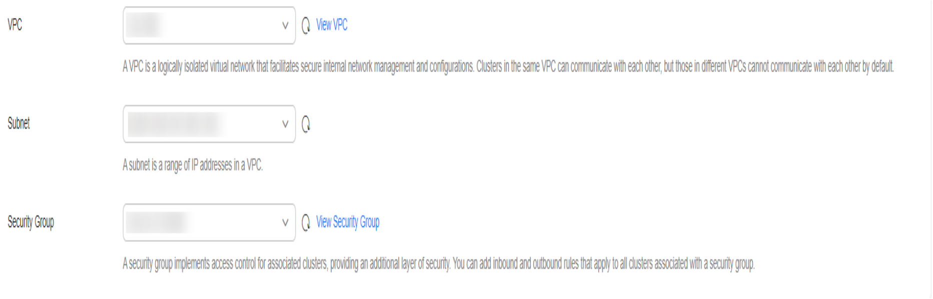
Parameter	Description	Example Value
Client node	Client nodes receive and coordinate external requests, such as search and write requests.	Unselect it.
Cold data node	Cold data nodes are used to store data that is not particularly sensitive to query latency in large quantities.	Unselect it.

7. Set the enterprise project.

   When creating a CSS cluster, you can bind an enterprise project to the cluster if you have enabled the enterprise project function. In this example, **default**, the default enterprise project, is selected.

8. Click **Next: Network**. Configure the cluster network.

Figure 2-3 Configuring networking



A VPC is a logically isolated virtual network that facilitates secure internal network management and configurations. Clusters in the same VPC can communicate with each other, but those in different VPCs cannot communicate with each other by default.

A subnet is a range of IP addresses in a VPC.

A security group implements access control for associated clusters, providing an additional layer of security. You can add inbound and outbound rules that apply to all clusters associated with a security group.

Table 2-4 Network configuration parameters

Parameter	Description	Example Value
VPC	Specify a VPC to isolate the cluster's network. **NOTE** The VPC must contain CIDRs. Otherwise, cluster creation will fail. By default, a VPC will contain CIDRs.	vpc-default
Subnet	A subnet provides dedicated network resources that are isolated from other networks, improving network security.	subnet-default
Security Group	A security group serves as a virtual firewall that provides access control policies for clusters. **NOTE** For enable cluster access, ensure that port 9200 is allowed by the security group.	default

Parameter	Description	Example Value
Security Mode	After the security mode is enabled, communication will be encrypted and authentication required for the cluster.	Disable

9. Click **Next: Advanced Settings**. Configure automatic snapshot creation and other functions.

   This cluster is used only for getting started. Cluster snapshots and advanced functions are not required.

10. Click **Next: Confirm**. Check the configuration and click **Next** to create a cluster.

11. Click **Back to Cluster List** to switch to the **Clusters** page. The cluster you created is now in the cluster list and its status is **Creating**. If the cluster is successfully created, its status changes to **Available**.

Step 2: Importing Data

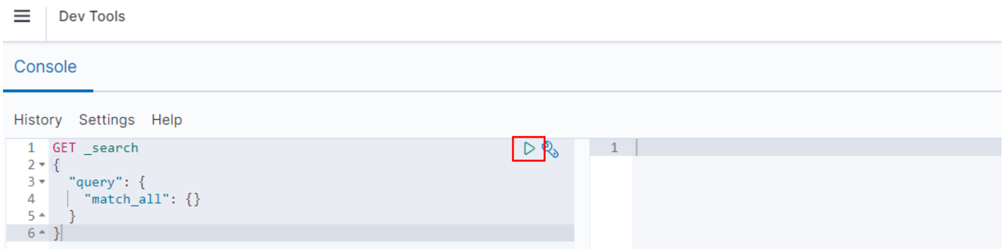
There are many ways to import data to an OpenSearch cluster. In this example, we use an open-source Elasticsearch API to import data on OpenSearch Dashboards.

1. On the OpenSearch cluster management page, select the created **Sample-OSCluster** cluster and click **Access Kibana** in the **Operation** column to access the OpenSearch Dashboards console.

2. In the OpenSearch Dashboards navigation pane on the left, choose **Dev Tools**.

   The text box on the left is the input box. The triangle icon in the upper right corner of the input box is the command execution button. The text box on the right area is the result output box.

Figure 2-4 Console page



```
GET _search
{
  "query": {
    "match_all": {}
  }
}
```

3. On the **Console** page, run the following command to create an index named **my_store**:

```
PUT /my_store
{
  "settings": {
    "number_of_shards": 1
  },
  "mappings": {
    "properties": {
      "productName": {
        "type": "text",
        "analyzer": "ik_smart"
      },
      "size": {
          "type": "keyword"
        }
      }
    }
  }
```

The command output is similar to the following:

```
{
  "acknowledged" : true,
  "shards_acknowledged" : true,
  "index" : "my_store"
}
```

4. On the **Console** page, run the following command to import data to the index named **my_store**:

```
POST /my_store/_doc/_bulk
{"index":{}}
{"productName":"Latest elegant shirts in autumn 2017","size":"L"}
{"index":{}}
{"productName":"Latest elegant shirts in autumn 2017","size":"M"}
{"index":{}}
{"productName":"Latest elegant shirts in autumn 2017","size":"S"}
{"index":{}}
{"productName":"Latest jeans in spring 2018","size":"M"}
{"index":{}}
{"productName":"Latest jeans in spring 2018","size":"S"}
{"index":{}}
{"productName":"Latest casual pants in spring 2017","size":"L"}
{"index":{}}
{"productName":"Latest casual pants in spring 2017","size":"S"}
```

If the value of the **errors** field in the command output is **false**, the data is imported successfully.

Step 3: Searching for Data

Perform full-text search and result aggregation and display in the OpenSearch cluster.

- **Full-text search**

  If you access the e-commerce website and want to search for items whose names include "spring jeans", enter "spring jeans" to begin your search.

  Run the following command on OpenSearch Dashboards:

```
GET /my_store/_search
{
  "query": {"match": {
    "productName": "spring jeans"
  }}
}
```

  The command output is similar to the following:

```
{
  "took" : 3,
  "timed_out" : false,
  "_shards" : {
    "total" : 1,
    "successful" : 1,
    "skipped" : 0,
    "failed" : 0
  },
  "hits" : {
    "total" : {
      "value" : 4,
      "relation" : "eq"
    },
    "max_score" : 1.7965372,
    "hits" : [
      {
        "_index" : "my_store",
        "_type" : "_doc",
        "_id" : "9xf6VHIBfClt6SDjw7H5",
        "_score" : 1.7965372,
        "_source" : {
          "productName": "Latest jeans in spring 2018",
          "size" : "M"
        }
      },
      {
        "_index" : "my_store",
        "_type" : "_doc",
        "_id" : "-Bf6VHIBfClt6SDjw7H5",
        "_score" : 1.7965372,
        "_source" : {
          "productName": "Latest jeans in spring 2018",
          "size" : "S"
        }
      },
      {
        "_index" : "my_store",
        "_type" : "_doc",
        "_id" : "-Rf6VHIBfClt6SDjw7H5",
        "_score" : 0.5945667,
        "_source" : {
          "productName": "Latest casual pants in spring 2017",
          "size" : "L"
        }
      },
      {
        "_index" : "my_store",
        "_type" : "_doc",
        "_id" : "-hf6VHIBfClt6SDjw7H5",
        "_score" : 0.5945667,
        "_source" : {
          "productName": "Latest casual pants in spring 2017",
          "size" : "S"
        }
      }
    ]
  }
}
```

- OpenSearch supports IK word segmentation. The command above segments "spring jeans" into "spring" and "jeans".
- OpenSearch supports full-text search. The command above searches for the information about all items whose names include "spring" or "jeans".
- Unlike traditional databases, OpenSearch can return results in milliseconds by using inverted indexes.
- OpenSearch supports ranking by score. In the command output, information about the first two items contains both "spring" and "jeans", while that about the last two items contain only "spring". Therefore, the first two items rank higher than the last two due to high keyword match.

- **Aggregated result display**

  The e-commerce website displays aggregated results. For example, it classifies items corresponding to "spring" based on sizes so that you can count the number of items of different sizes.

  Run the following result aggregation command on OpenSearch Dashboards:

```
GET /my_store/_search
{
"query": {
```

```
"match": { "productName": "spring" }
},
"size": 0,
"aggs": {
"sizes": {
"terms": { "field": "size" }
}
}
}
```

The command output is similar to the following:

```
{
  "took" : 3,
  "timed_out" : false,
  "_shards" : {
    "total" : 1,
    "successful" : 1,
    "skipped" : 0,
    "failed" : 0
  },
  "hits" : {
    "total" : {
      "value" : 4,
      "relation" : "eq"
    },
    "max_score" : null,
    "hits" : [ ]
  },
  "aggregations" : {
    "sizes" : {
      "doc_count_error_upper_bound" : 0,
      "sum_other_doc_count" : 0,
      "buckets" : [
        {
          "key" : "S",
          "doc_count" : 2
        },
        {
          "key" : "L",
          "doc_count" : 1
        },
        {
          "key" : "M",
          "doc_count" : 1
        }
      ]
    }
  }
}
```

Step 4: Deleting Indexes

If an index is no longer used, run the following command on OpenSearch Dashboards to delete the index to reclaim resources:

```
DELETE /my_store
```

The command output is similar to the following:

```
{
  "acknowledged" : true
}
```

Follow-up Operations

You can delete the cluster if you no longer need it.

NOTE

After you delete a cluster, its data cannot be restored. Exercise caution when deleting a cluster.

1. Log in to the CSS management console.
2. In the navigation pane on the left, choose **Clusters** > **OpenSearch**.
3. In the cluster list, locate the **Sample-OSCluster** cluster, and choose **More** > **Delete** in the **Operation** column.
4. In the confirmation dialog box, type in **DELETE**, and click **OK**.

3 Using Logstash for Data Migration

Logstash can be used to collect, transform, clean, and parse logs. This section offers an example of using a Logstash cluster to migrate data between different Elasticsearch clusters. Through this example, you can learn how to use the Logstash service, including creating clusters, importing and exporting data, and configuring tasks.

Procedure

The following describes how to use a Logstash cluster to migrate data from one Elasticsearch cluster to another.

Before starting to migrate data, make the necessary preparations. For details, see **Preparations**.

1. **Step 1: Obtaining Elasticsearch Cluster Information**: Obtain the addresses of the source and destination Elasticsearch clusters.
2. **Step 2: Creating a Logstash Cluster**: Create a Logstash cluster for migrating data between Elasticsearch clusters.
3. **Step 3: Configuring a Data Migration Task**: Configure an Elasticsearch cluster migration task for the Logstash cluster.
4. **Step 4: Starting the Migration Task**: Start the migration task in the Logstash cluster.
5. **Step 5: Stopping the Task**: After data migration is complete, stop the migration task.
6. **Step 6: Deleting the Cluster**: Delete clusters that you no longer need to reclaim resources.

Preparations

- You have registered with Huawei Cloud and performed real-name authentication. Make sure your account is not frozen or in arrears.

  If you do not have a Huawei Cloud account, perform the following operations to create one:

  a. Visit the **Huawei Cloud official website**.

  b. In the upper right corner of the page, click **Register** and complete the registration as prompted.

c. Select the service agreement and click **Enable**.

d. Perform real-name authentication.

- If your account is an individual account, see **Individual Real-Name Authentication**.
- If your account is an enterprise account, see **Enterprise Real-Name Authentication**.

- The source Elasticsearch cluster (**Source-ES**) and destination Elasticsearch cluster (**Dest-ES**) are ready. Both clusters are single-node non-security mode clusters.

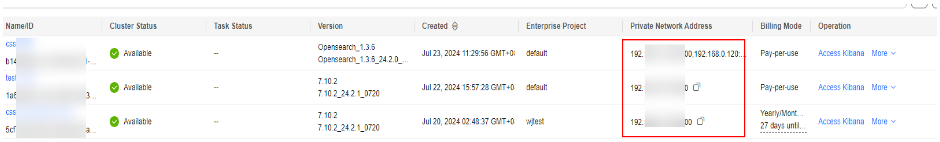
Step 1: Obtaining Elasticsearch Cluster Information

Obtain the addresses of the source and destination Elasticsearch clusters. For security-mode clusters, contact the administrator to obtain their usernames and passwords.

1. Log in to the CSS management console.
2. In the navigation pane on the left, choose **Clusters** > **Elasticsearch**.
3. In the cluster list, obtain the IP addresses of ElasticSearch clusters from the **Private Network Address** column. Generally, the IP address format is *<host>***:***<port>* or *<host>***:***<port>***,***<host>***:***<port>*.

   In this example, the address of the source Elasticsearch cluster (**Source-ES**) is **10.62.179.32:9200**, and that of the destination Elasticsearch cluster (**Dest-ES**) is **10.62.179.33:9200**.

Figure 3-1 Obtaining IP addresses



Name/ID	Cluster Status	Task Status	Version	Created	Enterprise Project	Private Network Address	Billing Mode	Operation
css[redacted] b14[redacted]l-...	Available	--	Opensearch_1.3.6 Opensearch_1.3.6_24.2.0_...	Jul 23, 2024 11:29:56 GMT+0:	default	192.[redacted]00,192.168.0.120:...	Pay-per-use	Access Kibana More
test[redacted] 1a6[redacted]3...	Available	--	7.10.2 7.10.2_24.2.1_0720	Jul 22, 2024 15:57:28 GMT+0	default	192.[redacted]0	Pay-per-use	Access Kibana More
css[redacted] 5cf[redacted]a...	Available	--	7.10.2 7.10.2_24.2.1_0720	Jul 20, 2024 02:48:37 GMT+0	wjtest	192.[redacted]00	Yearly/Mont... 27 days until...	Access Kibana More

Step 2: Creating a Logstash Cluster

Create a Logstash cluster for migrating data between Elasticsearch clusters.

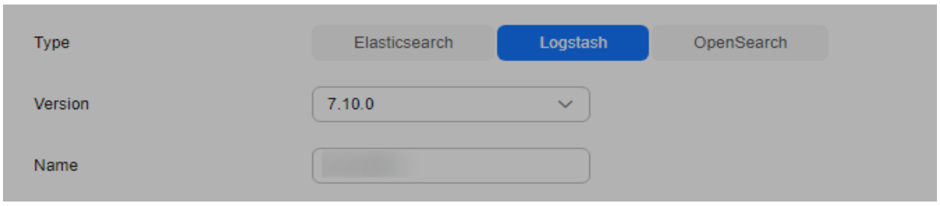
1. Log in to the CSS management console.
2. In the navigation pane on the left, choose **Clusters** > **Logstash**.
3. Click **Create Cluster** in the upper right corner. The **Create Cluster** page is displayed.
4. Configure Billing Mode and AZ for the cluster.

Table 3-1 Billing mode and AZ parameters

Parameter	Description	Example Value
Billing Mode	Select **Yearly/Monthly** or **Pay-per-use**. ● **Yearly/monthly**: You pay for the cluster by year or month, in advance. The service duration ranges from one month to three years. If you plan to use a cluster for more than nine months, you are advised to purchase a yearly package for a better price. A yearly package costs the same as a 10 monthly package. ● **Pay-per-use**: You are billed by actual duration of use, with a billing cycle of one hour. For example, 58 minutes of usage will be rounded up to an hour and billed.	Pay-per-use
Region	Select the region where the cluster is located. ECSs in different regions cannot communicate with each other over an intranet. For lower network latency and quicker resource access, select the nearest region.	Hong Kong, China
AZ	Select AZs associated with the cluster region. A maximum of three AZs can be configured.	AZ 1

5. Configure basic cluster information.

Figure 3-2 Configuring cluster information



Type: Elasticsearch | **Logstash** | OpenSearch

Version: 7.10.0

Name:

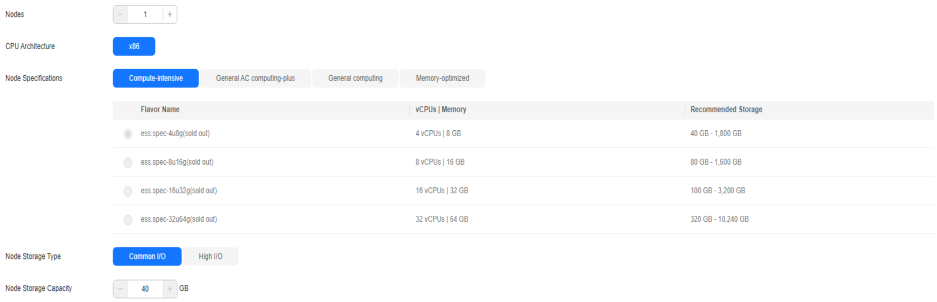
Table 3-2 Basic configuration parameters

Parameter	Description	Example Value
Cluster Type	Select **Logstash**.	Logstash

Parameter	Description	Example Value
Version	Select a cluster version from the drop-down list box.	7.10.0
Name	Cluster name, which contains 4 to 32 characters. Only letters, numbers, hyphens (-), and underscores (_) are allowed and the value must start with a letter.	Sample-Logstash

6. Configure the cluster's node specifications

Figure 3-3 Configuring the cluster's node specifications



Nodes: 1

CPU Architecture: x86

Node Specifications: Compute-intensive | General AC computing-plus | General computing | Memory-optimized

	Flavor Name	vCPUs | Memory	Recommended Storage
●	ess.spec-4u8g(sold out)	4 vCPUs | 8 GB	40 GB - 1,800 GB
○	ess.spec-8u16g(sold out)	8 vCPUs | 16 GB	80 GB - 1,600 GB
○	ess.spec-16u32g(sold out)	16 vCPUs | 32 GB	100 GB - 3,200 GB
○	ess.spec-32u64g(sold out)	32 vCPUs | 64 GB	320 GB - 10,240 GB

Node Storage Type: Common I/O | High I/O

Node Storage Capacity: 40 GB

Table 3-3 Specification parameters

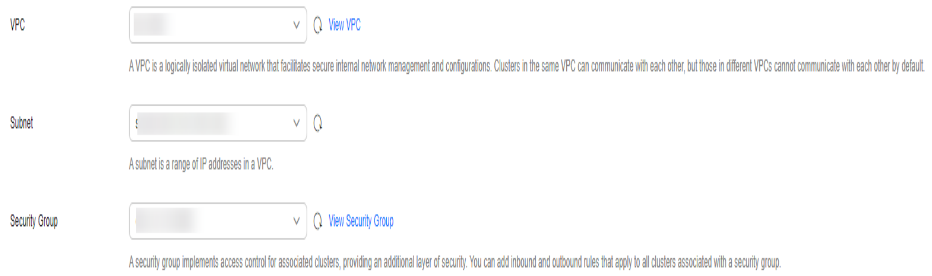
Parameter	Description	Example Value
Nodes	Number of nodes in a cluster, in the range 1 to 100.	1
CPU Architecture	Only **x86** is supported.	x86
Node Specifications	Select the specifications of cluster nodes.	ess.spec-4u8g
Node Storage Type	Select the storage type of cluster nodes.	High I/O
Node Storage Capacity	Set the storage capacity of a single cluster node. The default value is 40 GB.	40GB

7. Set the enterprise project.

   When creating a CSS cluster, you can bind an enterprise project to the cluster if you have enabled the enterprise project function. In this example, **default**, the default enterprise project, is selected.

8. Click **Next: Network** to configure the cluster network.

Figure 3-4 Configuring networking



VPC [redacted] View VPC

A VPC is a logically isolated virtual network that facilitates secure internal network management and configurations. Clusters in the same VPC can communicate with each other, but those in different VPCs cannot communicate with each other by default.

Subnet [redacted]

A subnet is a range of IP addresses in a VPC.

Security Group [redacted] View Security Group

A security group implements access control for associated clusters, providing an additional layer of security. You can add inbound and outbound rules that apply to all clusters associated with a security group.

Table 3-4 Network configuration parameters

Parameter	Description	Example Value
VPC	Specify a VPC to isolate the cluster's network. Select the VPC used by the Elasticsearch clusters. **NOTE** The VPC must contain CIDRs. Otherwise, cluster creation will fail. By default, a VPC will contain CIDRs.	vpc-default
Subnet	A subnet provides dedicated network resources that are isolated from other networks, improving network security.	subnet-default
Security Group	A security group serves as a virtual firewall that provides access control policies for clusters. **NOTE** For enable cluster access, ensure that port 9200 is allowed by the security group.	default

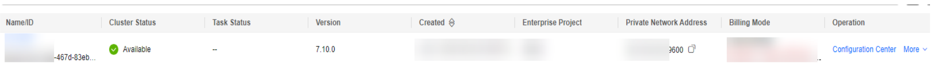
9. Click **Next: Configure Advanced Settings**.

   This cluster is used only for getting started. There is no need to enable advanced settings.

10. Click **Next: Confirm**. Check the configuration and click **Next** to create a cluster.

11. Click **Back to Cluster List** to switch to the **Clusters** page. The cluster you created is now in the cluster list and its status is **Creating**. If the cluster is successfully created, its status changes to **Available**.

Figure 3-5 Creating a cluster



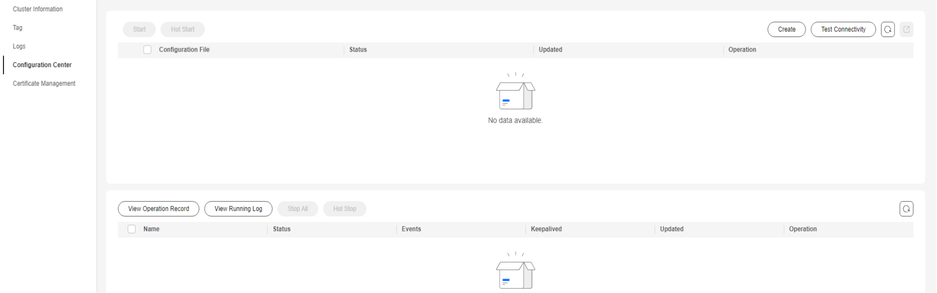
Name/ID	Cluster Status	Task Status	Version	Created	Enterprise Project	Private Network Address	Billing Mode	Operation
[redacted]-467d-83eb...	Available	--	7.10.0	[redacted]	[redacted]	[redacted]9600	[redacted]	Configuration Center More

Step 3: Configuring a Data Migration Task

Configure an Elasticsearch cluster migration task for the Logstash cluster.

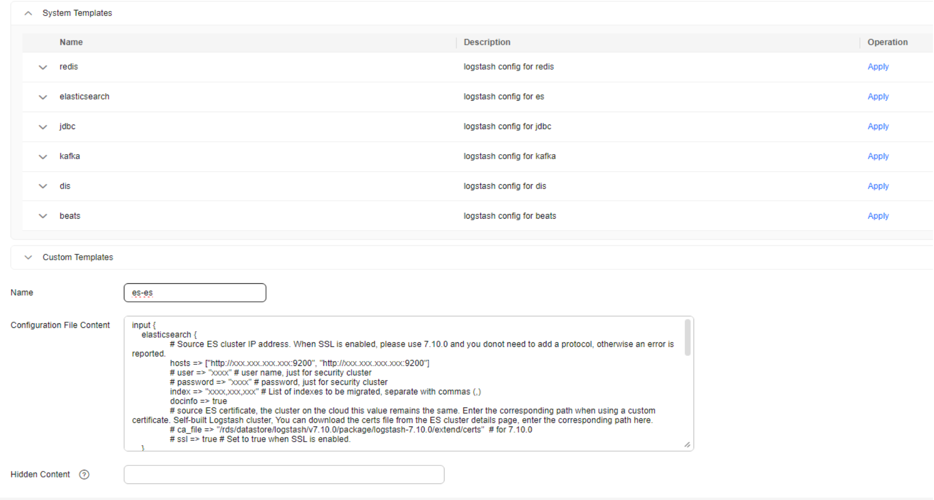
1. On the Logstash cluster management page, select the created **Sample-Logstash** cluster. The **Cluster Information** page is displayed.
2. Click **Configuration Center** on the right.

Figure 3-6 Logstash Configuration Center



3. On the Configuration Center page, click **Create** in the upper right corner. On the **Create Configuration File** page that is displayed, edit the configuration file.

Figure 3-7 Create Configuration File



System Templates

Name	Description	Operation
redis	logstash config for redis	Apply
elasticsearch	logstash config for es	Apply
jdbc	logstash config for jdbc	Apply
kafka	logstash config for kafka	Apply
dis	logstash config for dis	Apply
beats	logstash config for beats	Apply

Custom Templates

Name: es-es

Configuration File Content

```
input {
    elasticsearch {
            # Source ES cluster IP address. When SSL is enabled, please use 7.10.0 and you donot need to add a protocol, otherwise an error is reported.
            hosts => ["http://xxx.xxx.xxx.xxx:9200", "http://xxx.xxx.xxx.xxx:9200"]
            # user => "xxxx" # user name, just for security cluster
            # password => "xxxx" # password, just for security cluster
            index => "xxxx,xxx,xxx" # List of indexes to be migrated, separate with commas (,)
            docinfo => true
            # source ES certificate, the cluster on the cloud this value remains the same. Enter the corresponding path when using a custom certificate. Self-built Logstash cluster, You can download the certs file from the ES cluster details page, enter the corresponding path here.
            # ca_file => "/rds/datastore/logstash/v7.10.0/package/logstash-7.10.0/extend/certs"  # for 7.10.0
            # ssl => true # Set to true when SSL is enabled.
    }
```

Hidden Content

Table 3-5 Parameters for creating a configuration file

Parameter	Description	Example Value
Name	User-defined configuration file name. It can contain only letters, digits, hyphens (-), and underscores (_), and must start with a letter. The minimum length is 4 characters.	es-es

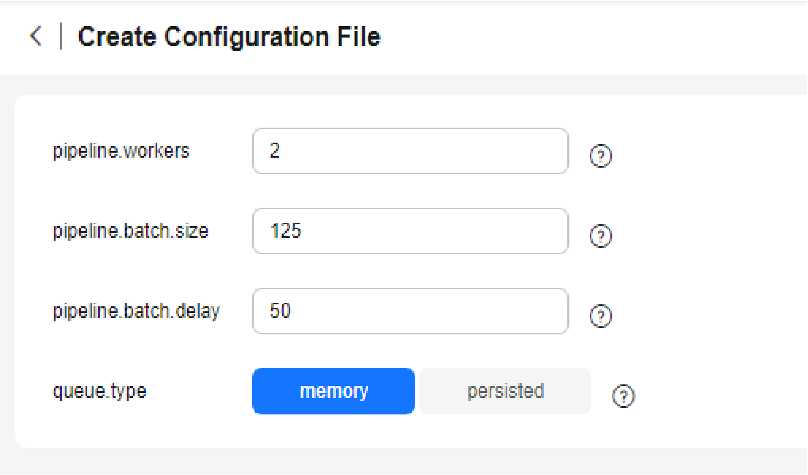
Parameter	Description	Example Value
Configuration File Content	Expand **System Templates**, find **elasticsearch**, and click **Apply** in the **Operation** column. In the **Configuration File Content** area, configure the configuration file based on comments in the template.	See **Table 3-6** for key configuration items. Use the default settings for others.
Hidden Content	For items that you enter in this box, the corresponding strings will be replaced with ***** in the configurations. Enter sensitive strings that you want to hide, and press **Enter**. You can enter a maximum of 20 strings, each with a maximum length of 512 bytes.	N/A

Table 3-6 Configuration item description

Configuration Item	Description	Example Value
hosts	Enter the addresses of the source and destination Elasticsearch clusters in **input** and **output**, respectively. For details about how to obtain the cluster addresses, see **Step 1: Obtaining Elasticsearch Cluster Information**.	input hosts: http://10.62.179.32:9200 output hosts: http://10.62.179.33:9200
user	Username for accessing the Elasticsearch cluster. This parameter is required for security-mode clusters. For non-security mode clusters, use **#** to comment out this parameter.	Use **#** to comment it out.
password	Password for accessing the Elasticsearch cluster. This parameter is required for security-mode clusters. For non-security mode clusters, use **#** to comment out this parameter.	Use **#** to comment it out.
index	Specifies indexes that need to be migrated. You can use a wildcard.	index*

4. Click **Next** to configure Logstash pipeline parameters.

Figure 3-8 Configuring pipeline parameters



Create Configuration File

pipeline.workers: 2

pipeline.batch.size: 125

pipeline.batch.delay: 50

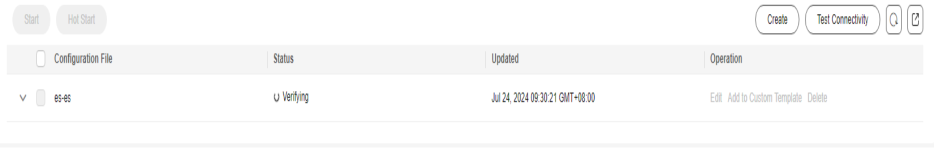
queue.type: memory | persisted

Table 3-7 Pipeline parameters

Parameter	Description	Example Value
pipeline.workers	Number of worker threads that will execute the Filters and Outputs stages of the pipeline in parallel.	4
pipeline.batch.size	Maximum number of events that a worker thread collects from inputs before attempting to execute its filters and outputs. A larger value is more effective but increases memory overhead.	125
pipeline.batch.delay	Maximum wait time for each new event before scheduling small batches to the pipeline worker thread and creating a pipeline event batch.	50
query.type	An internal queue model for event buffering. ● **memory** indicates a traditional memory-based queue. ● **persisted** indicates a disk-based ACKed persistent queue.	memory

5. Click Create. The system automatically verifies the configuration file. When the configuration file status changes to **Available**, the creation is successful.

Figure 3-9 Configuration file verification



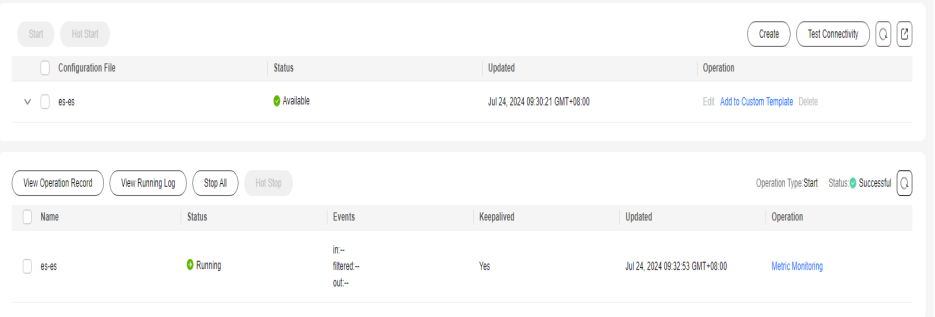
Configuration File	Status	Updated	Operation
es-es	Verifying	Jul 24, 2024 09:30:21 GMT+08:00	Edit Add to Custom Template Delete

Step 4: Starting the Migration Task

Start the configured migration task in the Logstash cluster.

1. On the Logstash cluster management page, select the created **Sample-Logstash** cluster. The **Cluster Information** page is displayed.
2. Click **Configuration Center** on the right.
3. Select a configuration file whose status is **Available**, and click **Start**. In the pipeline list, the **Events** column shows number of tasks processed by each stage of the pipeline.

Figure 3-10 Starting a task



Configuration File	Status	Updated	Operation
es-es	Available	Jul 24, 2024 09:30:21 GMT+08:00	Edit Add to Custom Template Delete

Name	Status	Events	Keepalived	Updated	Operation
es-es	Running	in:-- filtered:-- out:--	Yes	Jul 24, 2024 09:32:53 GMT+08:00	Metric Monitoring

4. After the data migration is complete, check the data consistency between the source and destination Elasticsearch clusters. For example, run the **_cat/indices** command in the source and destination clusters, separately, to check whether their indexes are consistent.
   a. On the Elasticsearch cluster management page, select the source Elasticsearch cluster **Source-ES** or the destination Elasticsearch cluster **Dest-ES**, and click **Access Kibana** in the **Operation** column to access the Kibana console.
   b. In the Kibana navigation pane on the left, choose **Dev Tools**.
   c. On the **Console** page, run the following command to view index information:

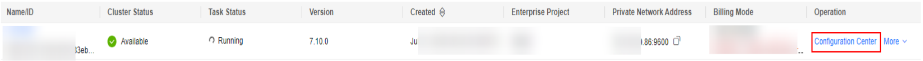
```
_cat/indices
```

Step 5: Stopping the Task

After data migration is complete, stop the migration task.

1. Log in to the CSS management console.
2. In the navigation pane on the left, choose **Clusters** > **Logstash**.
3. In the cluster list, select the **Sample-Logstash** cluster, and click **Configuration Center** in the **Operation** column.

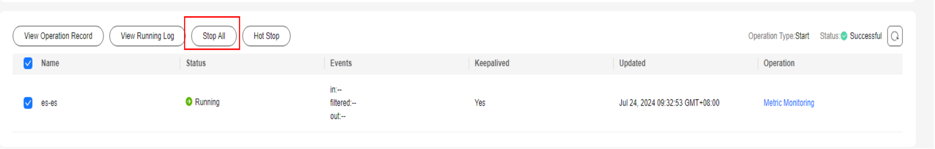
Figure 3-11 Configuration Center



Name/ID	Cluster Status	Task Status	Version	Created	Enterprise Project	Private Network Address	Billing Mode	Operation
[redacted]3eb...	Available	Running	7.10.0	Ju[redacted]	[redacted]	[redacted].86:9600	[redacted]	Configuration Center More

4. Select the name of the pipeline that has been started, and click Stop All to stop all running tasks. Wait until all pipeline tasks are stopped.

Figure 3-12 Stopping the task



Operation Type:Start Status: Successful

☑ Name	Status	Events	Keepalived	Updated	Operation
☑ es-es	Running	in:-- filtered:-- out:--	Yes	Jul 24, 2024 09:32:53 GMT+08:00	Metric Monitoring

Step 6: Deleting the Cluster

After data migration is completed, you may delete clusters created earlier to reclaim resources. Before you start, make sure that all pipeline tasks are stopped.

NOTE

Before deleting a cluster, stop all running tasks, and back up the necessary files.

1. Log in to the CSS management console.
2. In the navigation pane on the left, choose **Clusters** > **Logstash**.
3. In the cluster list, locate the **Sample-Logstash** cluster, and choose **More** > **Delete** in the **Operation** column.
4. In the confirmation dialog box, type in **DELETE**, and click **OK**.

4 Getting Started Through Common Practices

CSS provides you with best practices in different service scenarios and solution architectures, helping you select a proper practice solution based on your service scenario.

Table 4-1 Common practices

Scenario		Description
Migrating data	**Migrating Data Between Elasticsearch Clusters Using Huawei Cloud Logstash**	Huawei Cloud Logstash is a fully managed data ingestion and processing service. It is compatible with open-source Logstash and can be used for data migration between Elasticsearch clusters. You can use Huawei Cloud Logstash to migrate data from Huawei Cloud Elasticsearch, in-house built Elasticsearch, or third-party Elasticsearch to Huawei Cloud Elasticsearch.
	Migrating Data Between Elasticsearch Clusters Using Backup and Restoration	● **Migrating Data Between Huawei Cloud Elasticsearch Clusters Using Backup and Restoration**: Use this method only when both the source and destination clusters are CSS clusters that depend on OBS for storage service. ● **Migrating Data from an On-premises Elasticsearch Cluster to Huawei Cloud Using the S3 Plugin**: The S3 plugin can be used to migrate data from an on-premises Elasticsearch cluster to a Huawei Cloud Elasticsearch cluster. ● **Migrating Data from a Third-Party Elasticsearch Cluster to Huawei Cloud Using Backup and Restoration**: This option requires a storage repository.

Scenario		Description
	Migrating Data Between Huawei Cloud Elasticsearch Clusters Using the Read/Write Splitting Plugin	The read/write splitting plugin can be used for data migration only if both the source and destination clusters were created in CSS. It enables near-real time synchronization of index data between Elasticsearch clusters.
	Migrating data between Elasticsearch clusters using the CDM service	**Migrating the Entire Elasticsearch Database to CSS**: Huawei Cloud CDM can be used to migrate clusters between different cloud services.
	Migrating data from a Huawei Cloud Elasticsearch Cluster to an OpenSearch cluster	**Upgrading the Version of an Elasticsearch Cluster**: This method upgrades an Elasticsearch 7.10.2 cluster to an OpenSearch 1.3.6 cluster in CSS. CSS supports cross-engine cluster upgrades.
	Migrating data from a database to an Elasticsearch cluster	**Using DRS to Import Data from a Database to Elasticsearch**: Use this method to synchronize data from a MySQL database to an Elasticsearch cluster. Typically, this method is used for incremental data migration between clusters. After data synchronization, the clusters can be migrated at any time. This allows for more flexibility over other incremental migration solutions.
Accessing a cluster	**Elasticsearch Cluster Access Methods**	There are many ways to access an Elasticsearch cluster, such as Kibana, Cerebro, cURL commands, and Java, Python, Go, and MRS Hive clients.
	OpenSearch Cluster Access Methods	There are many ways to access an OpenSearch cluster, such as Kibana, Cerebro, and cURL commands.
Optimizing cluster performance	**Optimizing Write Performance**	Before using a CSS cluster, you are advised to optimize the write performance of the cluster to improve efficiency.
	Optimizing Query Performance	Before using a CSS cluster, you are advised to optimize the query performance of the cluster to improve efficiency.
Accelerating relational database query	**Using CSS to Accelerate Database Query and Analysis**	Elasticsearch is used as a supplement to relational databases, such as MySQL and GaussDB(for MySQL), to improve the full-text search and high-concurrency ad hoc query capabilities of the databases.

Scenario		Description
Building a log management platform	**Using CSS to Build a Unified Log Management Platform**	A unified log management platform built using CSS can manage logs in real time in a unified and convenient manner, enabling log-driven O&M and improving service management efficiency.
Sorting search results by custom rules	**Ranking Search Results Using Elasticsearch Custom Rules**	This practice describes how to configure custom rules to rank search results.