

Cloud Search Service

Troubleshooting

Issue 01
Date 2022-08-02



Copyright © Huawei Cloud Computing Technologies Co., Ltd. 2022. All rights reserved.

No part of this document may be reproduced or transmitted in any form or by any means without prior written consent of Huawei Cloud Computing Technologies Co., Ltd.

Trademarks and Permissions



HUAWEI and other Huawei trademarks are trademarks of Huawei Technologies Co., Ltd.

All other trademarks and trade names mentioned in this document are the property of their respective holders.

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 Clusters	1
1.1 Failed to Open Kibana	1
1.2 How Can I Improve Filebeat Performance?	2
1.3 Why Do I Fail to Access CSS Using TransportClient?	3
1.4 How Do I Handle the Error "Connection reset by peer" That Occurs When Spring Boot Uses ES?	3
1.5 What Do I Do If My Cluster Status Is Unavailable?	5
1.6 Why Does Cluster Creation Fail?	8
2 Data Import and Export	9
2.1 What Do I Do If Logs Cannot Be Written to CSS Due to High CPU Usage?	9
2.2 What Do I Do If an Error Is Reported When the Logstash on ECS Pushes Data to CSS?	10
3 Functions	11
3.1 Why Does Index Backup Fail?	11
3.2 What Do I Do If the Snapshot Repository Cannot Be Found?	11
3.3 What Do I Do If a Cluster Is Always in the Snapshot Creation State?	12
3.4 How Do I Back Up a Large Amount of Data Using Snapshots?	13
4 Ports	15
4.1 Why Does Access to Port 9200 Fail?	15
5 Change History	17

1 Clusters

1.1 Failed to Open Kibana

Symptom

After I click **Access Kibana** in the **Operation** column in the row where cluster **Es-event** resides on the **Clusters** page of the CSS management console, the Kibana page fails to be loaded and access to Kibana fails.

Cause

The browser cache is not cleared.

Procedure

1. Log in to the CSS management console.
2. In the left navigation pane, click **Clusters**.
3. On the displayed **Clusters** page, locate the row containing the target cluster **Es-event** and click **Access Kibana** in the **Operation** column.

NOTE

If the cluster has the security mode enabled, enter the username and password you for login. Generally, the username is **admin** and the password is the one specified during cluster creation.

If you forget the password, you can reset it on the cluster details page and then log in. For details.

4. On the displayed **Kibana** page, press **F12**.
5. Click **Network**, right-click **data:image**, and choose **clear browser cache** from the shortcut menu. In the displayed dialog box, click **OK**. Close the Kibana window.
6. Switch to the **Clusters** page, locate the row that contains cluster **Es-event** and click **Access Kibana** in the **Operation** column.

1.2 How Can I Improve Filebeat Performance?

Symptom

Filebeat is a high-performance file collection tool. By default, one core is allocated to Filebeat, and it writes 1 MB data to Elasticsearch per second. However, in practice, when a large number of service logs are generated, Filebeat cannot promptly collect and write them to Elasticsearch. In this case, you can optimize parameter settings in the **filebeat.yml** file to improve the Filebeat performance.

Procedure

1. Optimize the parameters involved in **input** of the **filebeat.yml** configuration file.
Increase the value of **harvester_buffer_size** based on actual requirements. This parameter defines the buffer size used by every **harvester**.
harvester_buffer_size: 40,960,000
Increase the value of **filebeat.spool_size** based on actual requirements. This parameter defines the number of log records that can be uploaded by the **spooler** at a time.
filebeat.spool_size: 250,000
Adjust the value of **filebeat.idle_timeout** based on actual requirements. This parameter defines how often the **spooler** is flushed. After the **idle_timeout** is reached, the **spooler** is flushed regardless of whether the **spool_size** has been reached.
filebeat.idle_timeout: 1s
2. Optimize the parameters involved in **output.elasticsearch** in the **filebeat.yml** configuration file.
Set the value of **worker** to the number of Elasticsearch clusters based on actual requirements. This parameter indicates the number of Elasticsearch clusters. The default value is **1**.
worker: 1
Increase the value of **bulk_max_size** based on actual requirements. This parameter defines the maximum number of events to bulk in a single Elasticsearch bulk API index request. The default is **50**.
bulk_max_size: 15,000
Adjust the value of **flush_interval** based on the actual requirements. This parameter defines the number of seconds to wait for new events between two bulk API index requests. If **bulk_max_size** is reached before this interval expires, additional bulk index requests are made.
flush_interval: 1s

1.3 Why Do I Fail to Access CSS Using TransportClient?

Issue

CSS cannot be accessed using the Spring Data Elasticsearch method, and the error message "None of the configured nodes are available" is displayed.

Symptom

CSS cannot be accessed using the Spring Data Elasticsearch method, and an error message is reported.

Possible Causes

Generally, **cluster.name** needs to be configured when you access a cluster using TransportClient. The possible cause of access failure is that the **Settings.EMPTY** option is used or the setting is incorrect.

Procedure

For details on how to access clusters using a client.

1.4 How Do I Handle the Error "Connection reset by peer" That Occurs When Spring Boot Uses ES?

Issue

When Spring Boot uses ES RestHighLevelClient to connect to ES, the error "Connection reset by peer" is reported, the TCP connection is interrupted, and service data fails to be written.

Symptom

The TCP connection is interrupted, and service data fails to be written.

Possible Causes

There are many possible causes. For example, the connection was disabled; the firewall, switch, or VPN was faulty; the keepalive settings were incorrect; the connected server node was changed; or the network was unstable.

Procedure

- Method 1
Modify the timeout interval of RestHighLevelClient connection requests. The default value is 1000 ms. You can increase the value to 10000 ms.

```
RestClientBuilder builder = RestClient.builder(new HttpHost(endpoint, port))
    .setHttpClientConfigCallback(httpClientBuilder->
httpClientBuilder.setDefaultCredentialsProvider(credentialsProvider))
```

```
.setRequestConfigCallback(requestConfigBuilder ->
requestConfigBuilder.setConnectTimeout(10000).setSocketTimeout(60000));
return new RestHighLevelClient(builder);
```

Settings for a single request:

request.timeout(TimeValue.timeValueSeconds(60));

- Method 2

Create a timer in Spring Boot to periodically check for the keepalive signals of ES.

```
@Scheduled(fixedRate = 60000, initialDelay = 60000)
public void keepConnectionAlive() {
    log.debug("Trying to ping Elasticsearch");
    try {
        final long noOfSportsFacilities = restHighLevelClient.status();
        log.debug("Ping succeeded for SportsFacilityViewRepository, it contains {} entities",
noOfSportsFacilities);
    } catch (Exception e) {
        log.debug("Ping failed for SportsFacilityViewRepository");
    }
}
```

- Method 3

Set the RestHighLevelClient keepalive time to 15 minutes.

- Method 4

Capture the exception in code and retry the request.

Reference

- TCP connections

TCP connections are classified into persistent connections and short connections. A short TCP connection is automatically disconnected after data packets are sent. A persistent TCP connection uses the keepalive timer function, and remains open for a certain period of time after data packets are sent.

- TCP keepalive mechanism

The keepalive mechanism is implemented using a timer. If the timer is activated, the server will send a keepalive probe packet. An ACK message is expected as a response. If the client does not respond, the server will terminate the connection. If the client responds, the keepalive timer will be reset.

The keepalive duration on the server is set to **30m**. In Linux, three parameters can be used to control the keepalive duration: **tcp_keepalive_time** (idle duration for enabling the keepalive function), **tcp_keepalive_intvl** (interval for sending keepalive packets), and **tcp_keepalive_probes** (the number of times the keepalive packets are sent if no response is received).

- http-keepalive

The http-keepalive mechanism enables a TCP connection to transmit as many packets as possible. The http-keepalive duration is updated each time a packet is transmitted. If the http-keepalive duration expires, it indicates that the client and server did not exchange packets during this period. In this case, the connection is automatically closed and released.

The tcp-keepalive mechanism retains a TCP connection until the connection is deliberately closed.

1.5 What Do I Do If My Cluster Status Is Unavailable?

Symptom

A CSS cluster status is **Unavailable**.

Possible Causes

The CSS backend reports unavailable cluster status to the console. The possible causes are as follows:

- The cluster is abnormal or faulty.
- The cluster background status is **red**.

Procedure

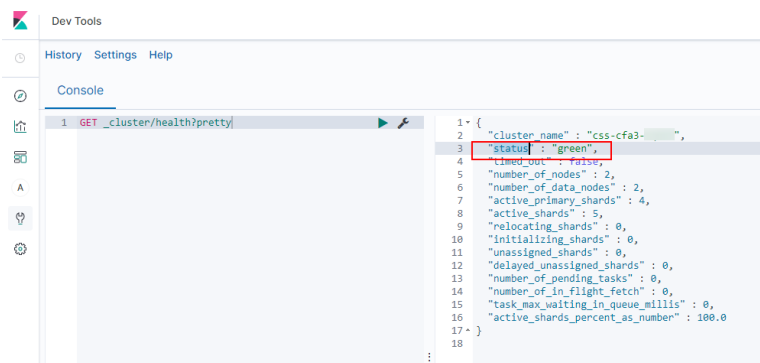
Check whether you can log in to Kibana.

If you can log in to Kibana, perform the following steps:

1. In the **Operation** column of the unavailable cluster, click **Access Kibana**.
2. In the navigation pane of Kibana, click **Dev Tools**.
3. Run the following command in **Dev Tools** to view the background status of the cluster:

```
GET _cluster/health?pretty
```

Figure 1-1 Viewing the cluster status



There are three possible background statuses of an Elasticsearch cluster:

- **green**: the cluster status is normal.
The background cluster status is checked once every minute, so the cluster status in the **Clusters** page is not updated in real time. You can wait for several minutes and check whether the cluster status changes to normal. If the status is still **Unavailable**, contact technical support.
- **yellow**: the replica shards of the cluster are abnormal.
initializing_shards indicates the number of shards that are being initialized. **unassigned_shards** indicates the number of shards that have not been allocated.

- **red**: the primary shards of the cluster are abnormal.
initializing_shards indicates the number of shards that are being initialized. **unassigned_shards** indicates the number of shards that have not been allocated.

Figure 1-2 Viewing shard information

```
1 {  
2   "cluster_name" : "css-cfa3-cqian",  
3   "status" : "green",  
4   "timed_out" : false,  
5   "number_of_nodes" : 2,  
6   "number_of_data_nodes" : 2,  
7   "active_primary_shards" : 4,  
8   "active_shards" : 5,  
9   "relocating_shards" : 0,  
10  "initializing_shards" : 0,  
11  "unassigned_shards" : 0,  
12  "delayed_unassigned_shards" : 0,  
13  "number_of_pending_tasks" : 0,  
14  "number_of_in_flight_fetch" : 0,  
15  "task_max_waiting_in_queue_millis" : 0,  
16  "active_shards_percent_as_number" : 100.0  
17 }
```

4. If there are shards being initialized, check whether the translog file is too large. When a primary shard is started, the translog file in the folder will be loaded automatically. A large translog file takes longer time for loading. Wait for about 10 minutes and check the cluster status again. If the status is still **Unavailable**, contact technical support.
5. If there are shards not allocated, perform the following steps:
 - a. Run the following command to check the reason:
GET /_cluster/allocation/explain?pretty
Possible reasons:
 - **INDEX_CREATED**: An API for creating an index is called. If the disk usage exceeds 85%, CSS will not assign new shards to the node. In this case, release storage space by referring to .
 - **CLUSTER_RECOVERED**: Full data restoration of the cluster is performed.
 - **INDEX_REOPENED**: An index is opened or closed.
 - **DANGLING_INDEX_IMPORTED**: The dangling index results are imported.
 - **NEW_INDEX_RESTORED**: Data is restored to a new index.
 - **EXISTING_INDEX_RESTORED**: Data is restored to disabled indexes.
 - **REPLICA_ADDED**: Replica shards are added explicitly.
 - **ALLOCATION_FAILED**: Shard assignment failed.
 - **NODE_LEFT**: The node that carries the shards is not in the cluster now.

- **REINITIALIZED:** Misoperations (such as using the shadow replica shard) were performed in the process from moving the shard to the shard initialization.
 - **REROUTE_CANCELLED:** The assignment is canceled because the routing is canceled explicitly.
 - **REALLOCATED_REPLICA:** A better replica location will be used, and the existing replica assignment is canceled.
- b. Run the following command to re-allocate shards:
- ```
POST /_cluster/reroute?retry_failed=true
```
- Wait for about 15 minutes. If the cluster status changes to **Available**, the fault has been rectified. Otherwise, perform the next step.
- c. If the shards are damaged and cannot be started, the shard reallocation failed. Run the following command to allocate an empty shard to the cluster:
- ```
POST /_cluster/reroute
{
  "commands": [
    {
      "allocate_empty_primary": {
        "index": "index-test", //Index name
        "shard": 13, //Index number
        "node": "css-test -ess-esn-11-1", //Node name
        "accept_data_loss": true
      }
    }
  ]
}
```
- Wait for about 15 minutes. If the cluster status changes to **Available**, the fault has been rectified. Otherwise, contact technical support.

If you cannot log in to Kibana, perform the following steps:

If a node is faulty, CSS first starts the node daemon process to rectify the fault. If the rectification fails, CSS will report that the node is unavailable.

The following faults may cause the rectification failure:

1. The network between nodes (for example, eth1 and eth1, and eth2 and eth2) is faulty. Nodes cannot ping each other.
Check the network between nodes.
2. Heavy cluster load causes nodes downtime frequently.
Locate the unavailable cluster and click **More > View Metric** in the **Operation** column to view its current and previous CPU, memory, and load usage. Check whether these metrics increased sharply or remained high for a long time. The surges may be caused by the sudden increase of access to the cluster. You can view the number of HTTP connections to learn about cluster access. Nodes with high load, CPU, or memory usage may go offline.
3. Too many shards (more than 50,000) exist, so the cluster cannot be started. When shards are started, the metadata related to the shards will be loaded to the memory. Too many shards require high memory. If a node goes offline or a new index is created, the master node has to use more computing resources to re-allocate such a large number of shards.

1.6 Why Does Cluster Creation Fail?

The following reasons may cause cluster creation to fail:

- Insufficient resource quota. You are advised to increase the resource quotas.
- The value of **Port Range/ICMP Type** in **Security Group** does not include port **9200**. Modify the security group information or select another available security group.
- For cluster version 7.6.2 and later versions, the communication port 9300 is enabled on the subnet of user VPC by default. When you create a cluster, check whether the selected security group allows traffic from communication port 9300 in the subnet. If it does not allow traffic, modify the security group or select another security group.

2 Data Import and Export

2.1 What Do I Do If Logs Cannot Be Written to CSS Due to High CPU Usage?

Issue

The CSS CPU usage is high, an error message "Elasticsearch Unreachable" is displayed on Logstash, and logs cannot be written to CSS.

Symptom

Logs cannot be written to CSS.

Possible Causes

The customer index has only one shard. The node of the shard is overloaded, and the job queue is full. Later jobs are rejected.

Procedure

1. Log in to the CSS management console.
2. Choose **Clusters** in the navigation pane. The cluster list is displayed.
3. Locate the target cluster and choose **More > Access Cerebro** in the **Operation** column.
If the cluster is in security mode, you need to enter the login username (**admin**) and password.
4. In Cerebro, view the number of shards in the cluster and metrics such as the CPU, load, head, and dis of each node.
5. Analyze the possible causes based on metrics and tune your system accordingly.
 - a. Increase the number of queues and reduce rejected jobs by changing the value of **write.queue_size**.

- i. Click the name of the target cluster whose parameters you want to modify. The basic information page of the cluster is displayed.
 - ii. Click **Parameter Configurations**, search for **write.queue_size**, and change its value.
If this parameter does not exist, add it in the **Customize** area. For details.
 - b. Rebuild the indexes to ensure that the number of shards is greater than that of nodes in the cluster.
6. If the number of shards and queues are appropriate but the CPU usage and load are still high, you are advised to scale out the cluster.

2.2 What Do I Do If an Error Is Reported When the Logstash on ECS Pushes Data to CSS?

Issue

After Logstash is deployed on an ECS, an error is reported when data is pushed to CSS.

Symptom

After Logstash is deployed on an ECS, an error is reported when data is pushed to CSS. The error message is as follows:

```
LogStash::Outputs::ElasticSearch::HttpClient::Pool::BadResponseCodeError: Got response code '500' contacting Elasticsearch at URL 'https://192.168.xx.xx:9200/_xpack'.
```

Possible Causes

CSS currently does not integrate the x-pack plugin. When you access CSS after deploying Logstash, the system will check whether x-pack is enabled for CSS.

Procedure

1. Delete the x-pack directory in Logstash.
2. Add the configuration item **ilm_enabled => false** to **elasticsearch** under the **output** tag in the Logstash configuration file.
3. Push data to CSS again.

3 Functions

3.1 Why Does Index Backup Fail?

Index backup is implemented by creating cluster snapshots. If index backup fails, perform the following steps to troubleshoot this problem:


Check Whether the Account or IAM User Has the Index Backup Permissions

1. Log in to the IAM management console.
2. Check the user group that the account or the IAM user belongs to.
For details, see "Viewing and Modifying User Information" in the *Identity and Access Management User Guide*.
3. Check whether the permissions assigned to the user group include the following two permissions: **Tenant Administrator** for project **OBS** in region **Global service** and **CSS Administrator** for the current region.
For details, see "Viewing and Modifying User Group Information" in the *Identity and Access Management User Guide*.
 - If neither of the preceding permissions has been assigned to the user group, go to [4](#).
 - If both the preceding permissions have been assigned to the group, contact technical support.
4. Add the following permissions to the user group: **Tenant Administrator** for project **OBS** in region **Global service**, and **CSS Administrator** for the current region.
For details, see "Viewing and Modifying User Group Information" in the *Identity and Access Management User Guide*.

3.2 What Do I Do If the Snapshot Repository Cannot Be Found?

1. On the **Clusters** page of the CSS management console, locate the target cluster and click **Access Kibana** in the **Operation** column.

- In the navigation pane of Kibana, click **Dev Tools**. Click **Get to work** to switch to the **Console** page.

Enter the code as required in the left pane, click  to execute the command, and view the result in the right pane.


- If no information is returned after the **GET _snapshot/_all** is executed, or if the error message shown in **Figure 3-1** is displayed after the **GET _snapshot/repo_auto/_all** command is executed, it indicates that no snapshot is configured. In this case, configure the snapshot again.

Figure 3-1 Returned information

```

1 {
2   "error": {
3     "root_cause": [
4       {
5         "type": "repository_missing_exception",
6         "reason": "[repo_auto] missing"
7       }
8     ],
9     "type": "repository_missing_exception",
10    "reason": "[repo_auto] missing"
11  },
12  "status": 404
13 }

```

- Click the name of the target cluster. On the displayed cluster details page, click the **Cluster Snapshots** tab.
- Click  next to **Basic Configuration** to modify the basic configurations.
- After the modification is complete, click **OK**.

If the repository still cannot be found after the modification, try modifying and restoring the backup path, and save the settings again.

3.3 What Do I Do If a Cluster Is Always in the Snapshot Creation State?

Possible causes are as follows:

- The cluster is heavily loaded, and snapshot creation takes a long time.
The default snapshot creation speed of a single node is 40 MB/s. The speed will be lower if the cluster is busy. You can query the status of a snapshot by referring to preceding sections.
You can run the **GET _snapshot/repo_auto/snapshot-name** command to check the number of shards that are being backed up. You can also terminate snapshot creation via APIs.
Solution: Wait for the snapshot creation to complete, or terminate the task.
- Failed to update snapshot information.
Elasticsearch stores ongoing snapshot information in the cluster state. After a snapshot is created, its state needs to be updated, but Elasticsearch may fail

to update the snapshot state due to high memory usage. Elasticsearch does not retry failed updates, so the snapshot remains in the Creating state.

Solution: Call the snapshot deletion API.

- Temporary AKs or SKs expire.

CSS uses an agency to write data in Elasticsearch to OBS. To create a snapshot repository, you need to use the agency to obtain a temporary AK and a temporary SK, and configure them in the repository. Temporary AKs and SKs have a validity period (24 hours). Snapshot creation will fail if it does not complete within 24 hours. In this case, the repository cannot be updated, queried, and deleted, and the cluster state information cannot be deleted manually or by a rolling restart. To delete residual snapshot information, perform a normal restart.

Solution: Currently, residual snapshot information can only be deleted in a normal restart. CSS will provide a termination interface to rectify the fault.

3.4 How Do I Back Up a Large Amount of Data Using Snapshots?

Improve snapshot backup configurations to ensure that each snapshot takes less than 24 hours to create. For example:

1. Specify indexes and back up data in batches. The default value is *, indicating that all indexes are backed up.
2. Use a custom snapshot repository.
 - a. Create a custom repository.

CSS provides the **repo_auto** repository by default. You can create one by calling the following API:

```
PUT _snapshot/my_backup
{
  "type": "obs",
  "settings": {
    "bucket": "css-backup-name", // Bucket name
    "base_path": "css_backup/711/", // Backup path
    "chunk_size": "2g",
    "endpoint": "obs.xxx.huawei.com:443", //OBS domain name address
    "region": "xxx", //Region name
    "compress": "true",
    "access_key": "xxxxx", //AK
    "secret_key": "xxxxxxxxxxxxxxxx" //SK
    "max_restore_bytes_per_sec": "100mb", // OBS speed. The default value is 40 MB. You
    can increase the value if your cluster can achieve higher performance.
    "max_snapshot_bytes_per_sec": "100mb"
  }
}
```

- b. Create a snapshot using a custom repository.

```
PUT _snapshot/my_backup/Snapshot_name
{
  "indices": "*", // Backup index. The asterisk (*) indicates indexes. Multiple indexes are
  separated by commas (.).
  "ignore_unavailable": true, // Whether to ignore the availability of a single index. The value
  true indicates that the availability is ignored.
  "include_global_state": false //: The default value is false, indicating that the cluster state and
  some other states are not saved.
}
```


c. Query the snapshot status.

```
GET _snapshot/my_backup/snapshot_name/_status
```

d. Restore indexes in the custom repository.

```
POST /_snapshot/my_backup/snapshot_name/_restore
```

```
{  
  "indices": "test-0000000000",  
  "ignore_unavailable": true,  
  "include_global_state": false,  
  "rename_pattern": "(.+)",  
  "rename_replacement": "$1"  
}
```

4 Ports

4.1 Why Does Access to Port 9200 Fail?

Symptom

If a VPN or VPC peering connection is used to access the CSS cluster, no result is returned when the curl command is used to connect to the CSS cluster.

For example, if you run the following command to connect to the cluster, no result is returned:

```
curl -s 'http://< node private access address >:9200'
```

Cause

If a VPN or VPC peering connection is used to access CSS, that means that the client and CSS are not in the same VPC. Therefore, the subnet of the CSS cluster must be in a different network segment from that of the VPC.

Suppose, for example, there is a CSS cluster in VPC **vpc-8e28** on the network segment **192.168.0.0/16**, the subnet **subnet-4a81** of the VPC is selected, and its network segment is also **192.168.0.0/16**. As the CSS subnet **vpc-8e28** and the subnet it is being accessed from (**subnet-4a81**) are both 192.168.0.0/16, if the VPN or the VPC peering connection tries to access the CSS cluster, the host created on the subnet does not have a gateway corresponding to the VPC. As a result, the default route of the CSS service is affected and access to port 9200 fails.

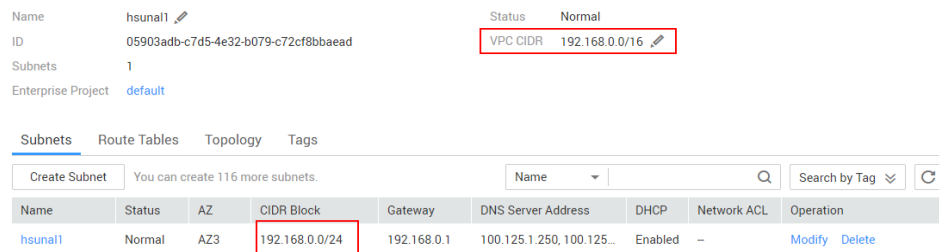
Procedure

If access to port 9200 fails but the CSS cluster is available, do as follows:

1. Go to the CSS management console. In the cluster list, click the cluster name to view the VPC and subnet used by the cluster.
2. Go to the VPC management console. In the VPC list, click the name of the VPC used by the CSS cluster. The VPC details page is displayed. View the VPC and subnet network segment information.

As shown in **Figure 4-1**, the VPC network segment information is the same as the subnet network segment information. When a VPN private line or a VPC peer connection is used, access to port 9200 fails.

Figure 4-1 Viewing network segment information



3. If the preceding error occurs, create another cluster and this time select a subnet that is different from the VPC subnet. If the subnet does not exist, create another subnet on the VPC management console.

After a new CSS cluster is created, migrate the data of the old cluster to the new cluster, and then use the VPN or VPC peering connection to access the cluster.

NOTE

If you require a VPN connection or VPC peering connection to access the CSS cluster, ensure that the VPC and subnet of the newly created CSS are in different network segments.

5 Change History

Release Date	Description
2022-08-02	This is the first official release.