Tencent Cloud TCHouse-D

Getting Started

Product Documentation

Contents

Getting Started

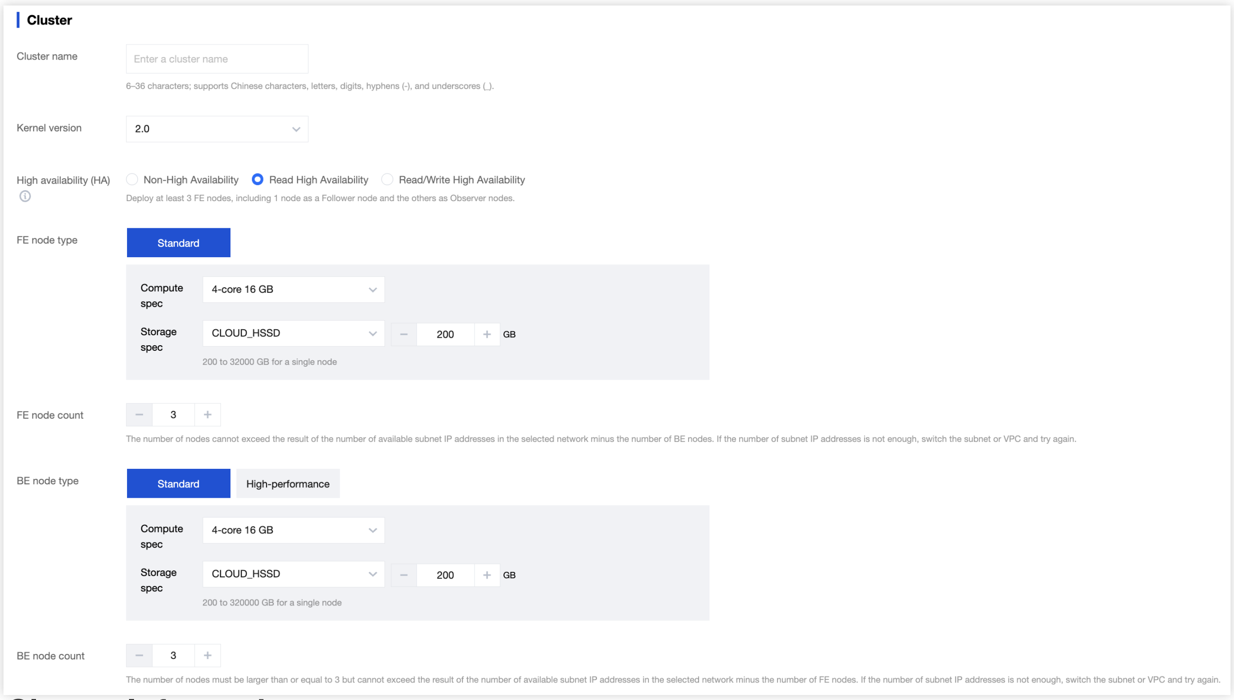
Using Tencent Cloud TCHouse-D Through the Console

Last updated：2024-06-27 10:39:42

This document describes how to use Tencent Cloud TCHouse-D through the console to create, manage, and execute SQL commands in the SQL workspace.

Creating Cluster

1. On the Tencent Cloud TCHouse-D introduction page, click **Buy Now**. Alternatively, log in to the Tencent Cloud TCHouse-D console and click **Create Cluster** to go to the purchase page for configuration and purchase.
2. After completing the basic configuration, cluster configuration, and log configuration, the system will display the corresponding configuration costs. Click **Activate** to create a cluster.



Cluster

Cluster name: Enter a cluster name
6–36 characters; supports Chinese characters, letters, digits, hyphens (-), and underscores (_).

Kernel version: 2.0

High availability (HA): Non-High Availability [✓ Read High Availability] Read/Write High Availability
Deploy at least 3 FE nodes, including 1 node as a Follower node and the others as Observer nodes.

FE node type: [✓ Standard]
Compute spec: 4-core 16 GB
Storage spec: CLOUD_HSSD 200 GB
200 to 32000 GB for a single node

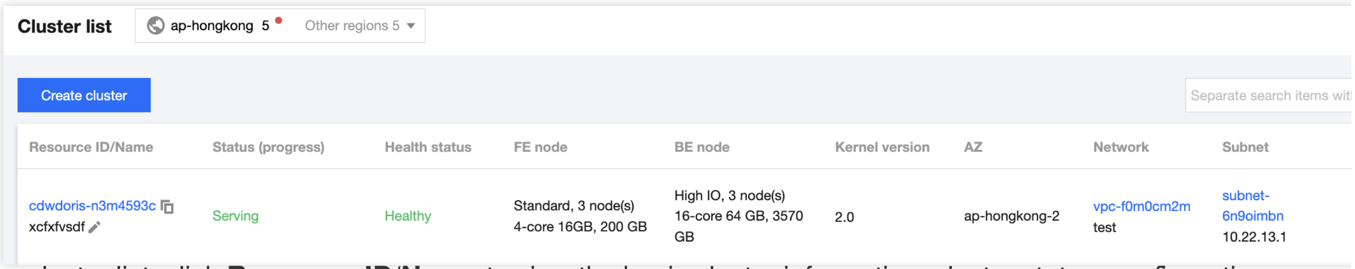
FE node count: 3
The number of nodes cannot exceed the result of the number of available subnet IP addresses in the selected network minus the number of BE nodes. If the number of subnet IP addresses is not enough, switch the subnet or VPC and try again.

BE node type: [✓ Standard] High-performance
Compute spec: 4-core 16 GB
Storage spec: CLOUD_HSSD 200 GB
200 to 320000 GB for a single node

BE node count: 3
The number of nodes must be larger than or equal to 3 but cannot exceed the result of the number of available subnet IP addresses in the selected network minus the number of FE nodes. If the number of subnet IP addresses is not enough, switch the subnet or VPC and try again.

View Cluster Information

1. After the cluster is created, you can access the Tencent Cloud TCHouse-D console and select the region where the cluster is located to view the status information of all clusters in that region from the cluster list.

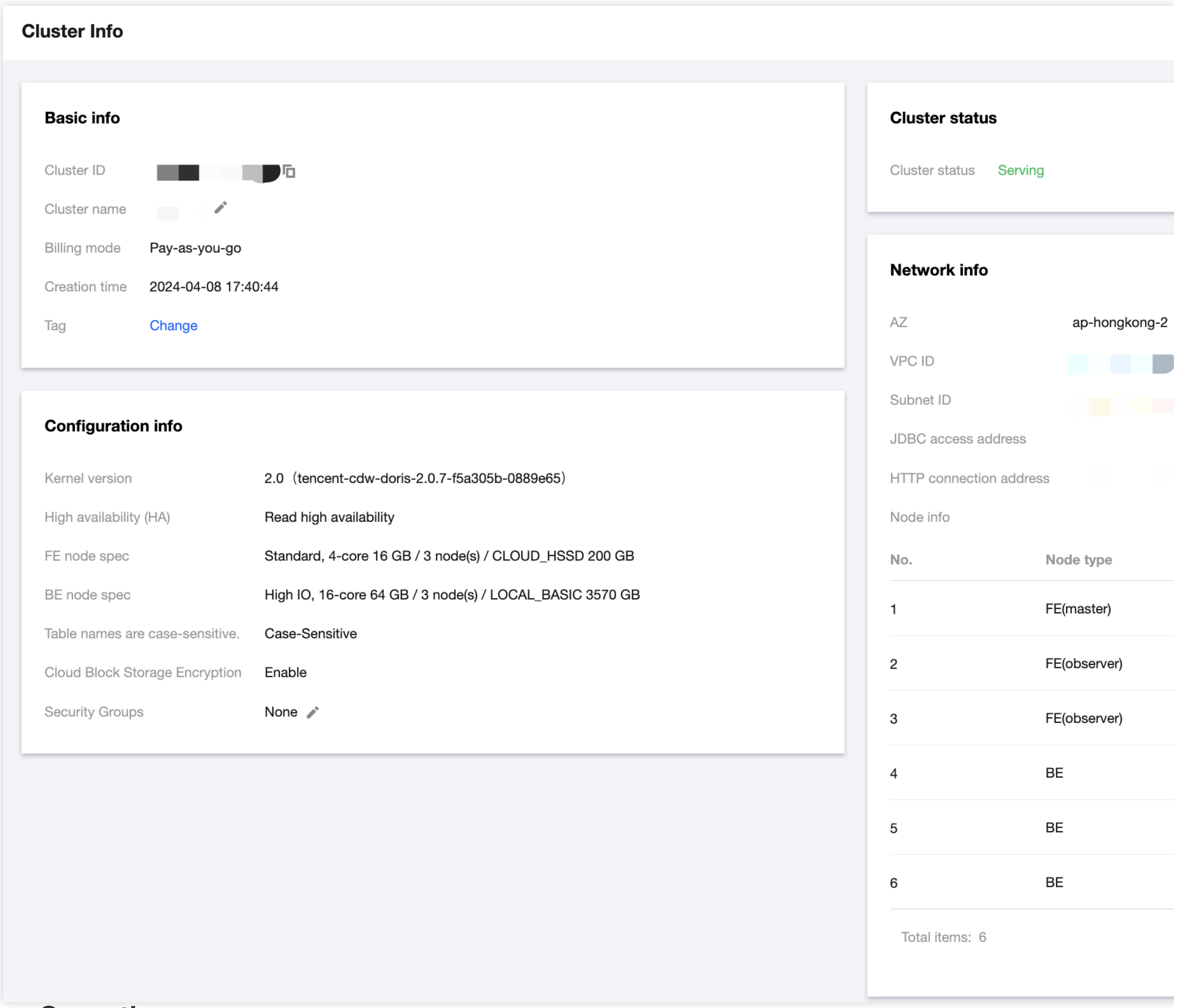


Cluster list | ap-hongkong 5 | Other regions 5

Create cluster

Resource ID/Name	Status (progress)	Health status	FE node	BE node	Kernel version	AZ	Network	Subnet
cdwdoris-n3m4593c xcfxfvsdf	Serving	Healthy	Standard, 3 node(s) 4-core 16GB, 200 GB	High IO, 3 node(s) 16-core 64 GB, 3570 GB	2.0	ap-hongkong-2	vpc-f0m0cm2m test	subnet-6n9oimbn 10.22.13.1

2. In the cluster list, click **Resource ID**/**Name** to view the basic cluster information, cluster status, configuration information, and network information.



Cluster Info

Basic info

Cluster ID	[redacted]
Cluster name	
Billing mode	Pay-as-you-go
Creation time	2024-04-08 17:40:44
Tag	Change

Configuration info

Kernel version	2.0（tencent-cdw-doris-2.0.7-f5a305b-0889e65）
High availability (HA)	Read high availability
FE node spec	Standard, 4-core 16 GB / 3 node(s) / CLOUD_HSSD 200 GB
BE node spec	High IO, 16-core 64 GB / 3 node(s) / LOCAL_BASIC 3570 GB
Table names are case-sensitive.	Case-Sensitive
Cloud Block Storage Encryption	Enable
Security Groups	None

Cluster status

Cluster status: Serving

Network info

AZ	ap-hongkong-2
VPC ID	
Subnet ID	
JDBC access address	
HTTP connection address	
Node info	

No.	Node type
1	FE(master)
2	FE(observer)
3	FE(observer)
4	BE
5	BE
6	BE

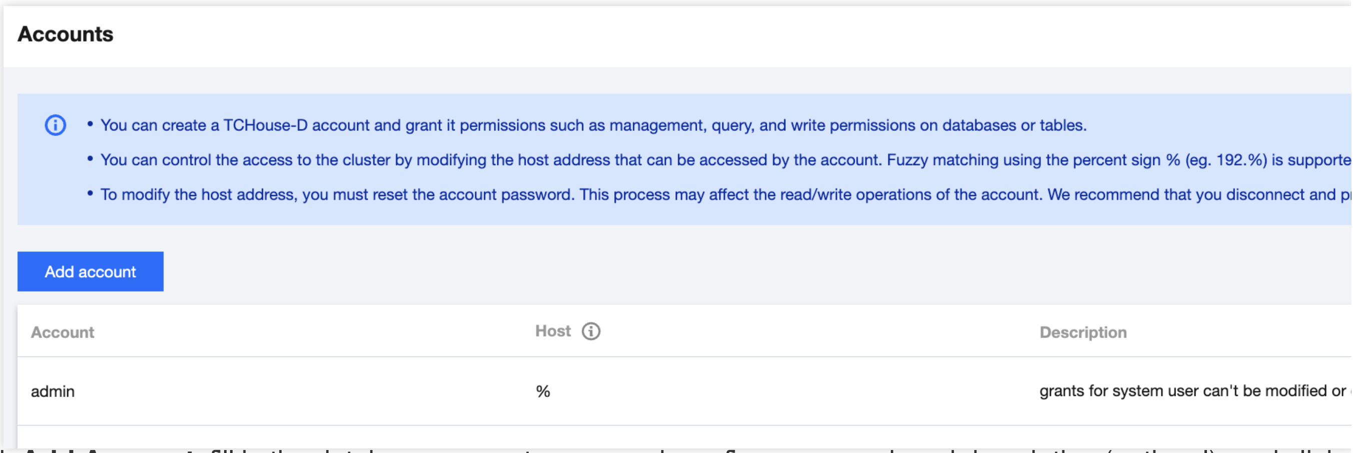
Total items: 6

Cluster Operation

In the console, you can conveniently manage account permissions, configure monitoring and alarms, modify configurations, and manage nodes. For details, see the Operation Guide section.

Managing Account Permissions

1. Tencent Cloud TCHouse-D offers a visual interface in the console, allowing for convenient and efficient account and permission management of clusters.



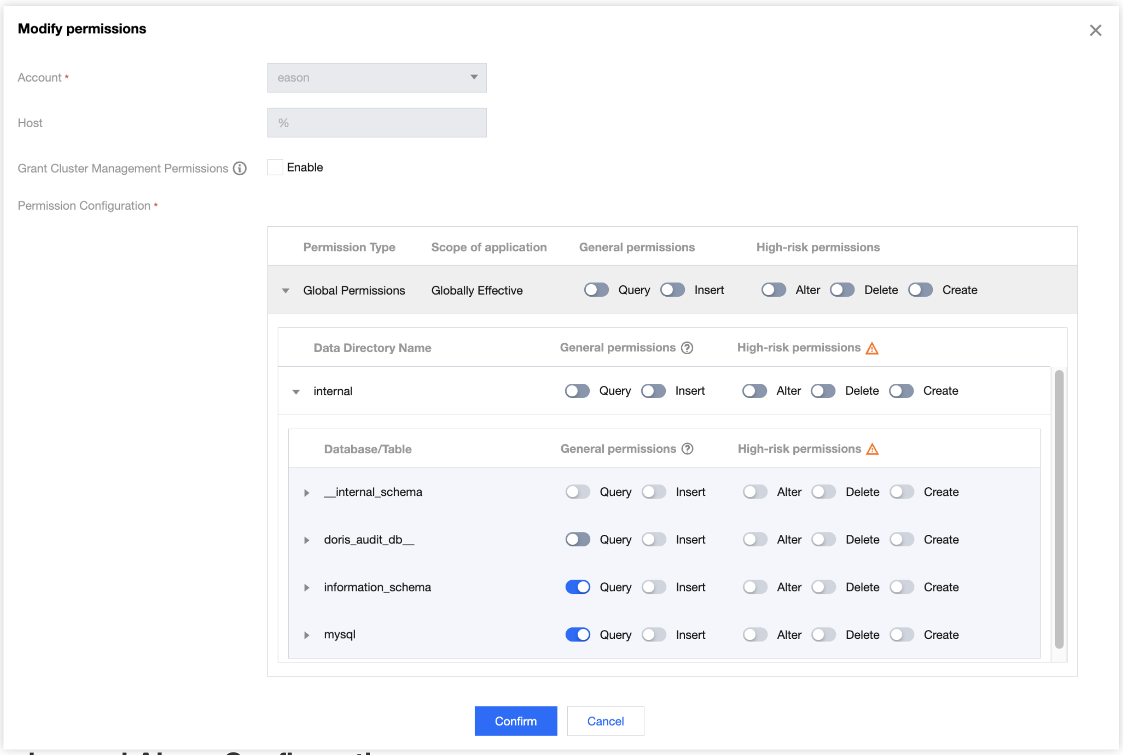
Accounts

- You can create a TCHouse-D account and grant it permissions such as management, query, and write permissions on databases or tables.
- You can control the access to the cluster by modifying the host address that can be accessed by the account. Fuzzy matching using the percent sign % (eg. 192.%) is supporte
- To modify the host address, you must reset the account password. This process may affect the read/write operations of the account. We recommend that you disconnect and pr

Add account

Account	Host	Description
admin	%	grants for system user can't be modified or

2. Click **Add Account**, fill in the database account, password, confirm password, and description (optional), and click **Confirm** to add an account. Click **Modify Permissions** to go to the permission modification window, where you can also view permissions.



Modify permissions

Account * eason

Host %

Grant Cluster Management Permissions ☐ Enable

Permission Configuration *

Permission Type	Scope of application	General permissions	High-risk permissions
Global Permissions	Globally Effective	Query Insert	Alter Delete Create

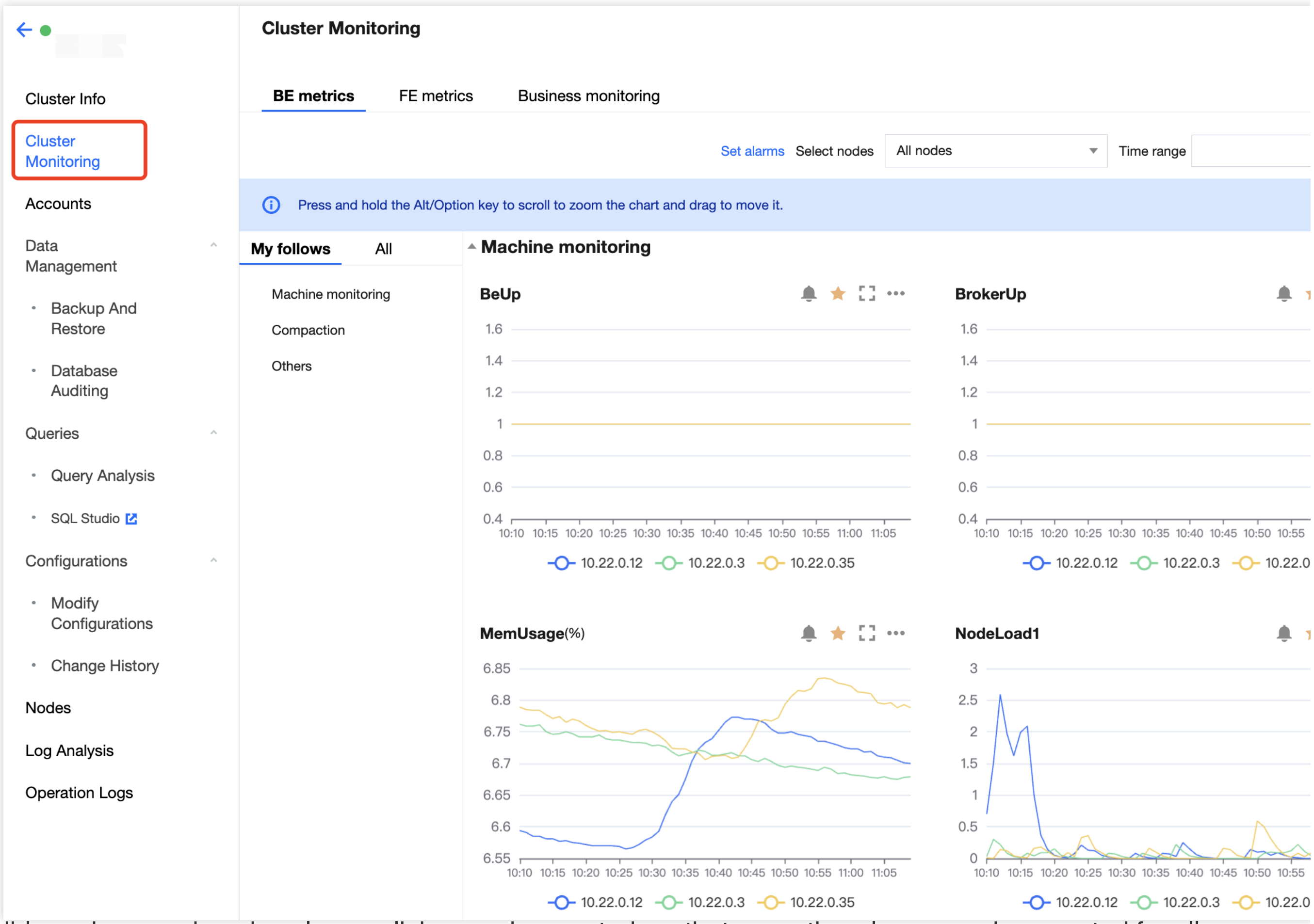
Data Directory Name	General permissions	High-risk permissions
internal	Query Insert	Alter Delete Create

Database/Table	General permissions	High-risk permissions
__internal_schema	Query Insert	Alter Delete Create
doris_audit_db__	Query Insert	Alter Delete Create
information_schema	Query Insert	Alter Delete Create
mysql	Query Insert	Alter Delete Create

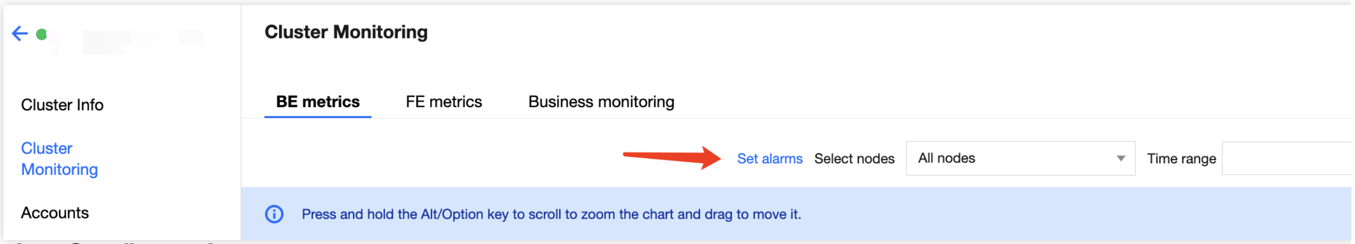
Confirm Cancel

Monitoring and Alarm Configuration

1. Tencent Cloud TCHouse-D provides a wealth of monitoring indicators to help you understand the operation status of the cluster. You can configure alarms for indicators to check operation information in real time, achieving rapid response. Click **Resource ID/Name** of the cluster to view cluster details, and click **Cluster Monitoring** to view the performance indicators of the cluster.

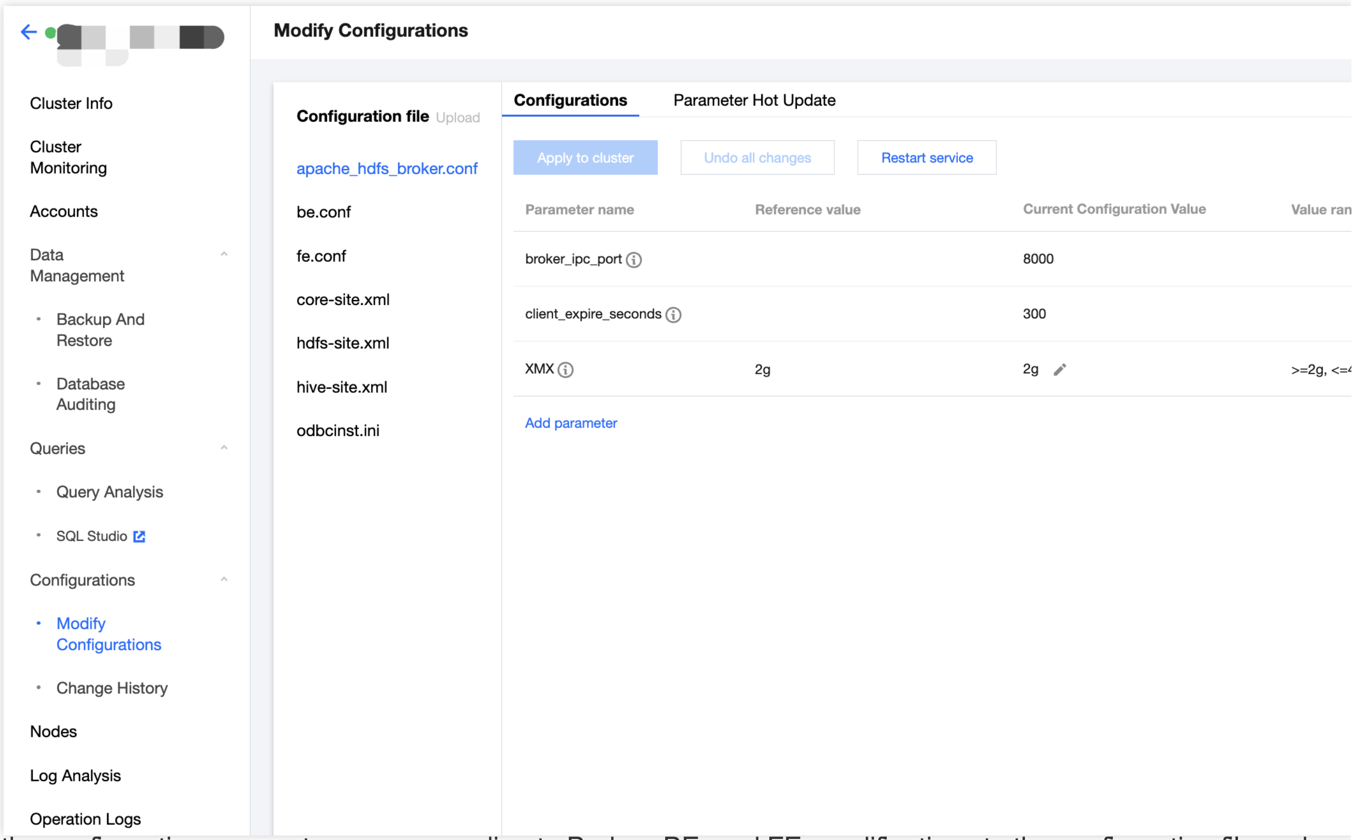


2. Flexible and comprehensive alarm policies can be created, so that exception alarms can be reported for all monitoring indicators. In the cluster monitoring page, click **Set Alarm** to create a new alarm policy.



Modifying Configuration

1. Tencent Cloud TCHouse-D supports flexible parameter configuration methods. You can comprehensively and intuitively set the parameters of BE/FE/Broker in the console, and view historical modification records. In the cluster list, click **Resource ID**/**Name** to view cluster details, and select **Configurations** > **Modify Configurations** from the left list to go to the parameter configuration page.

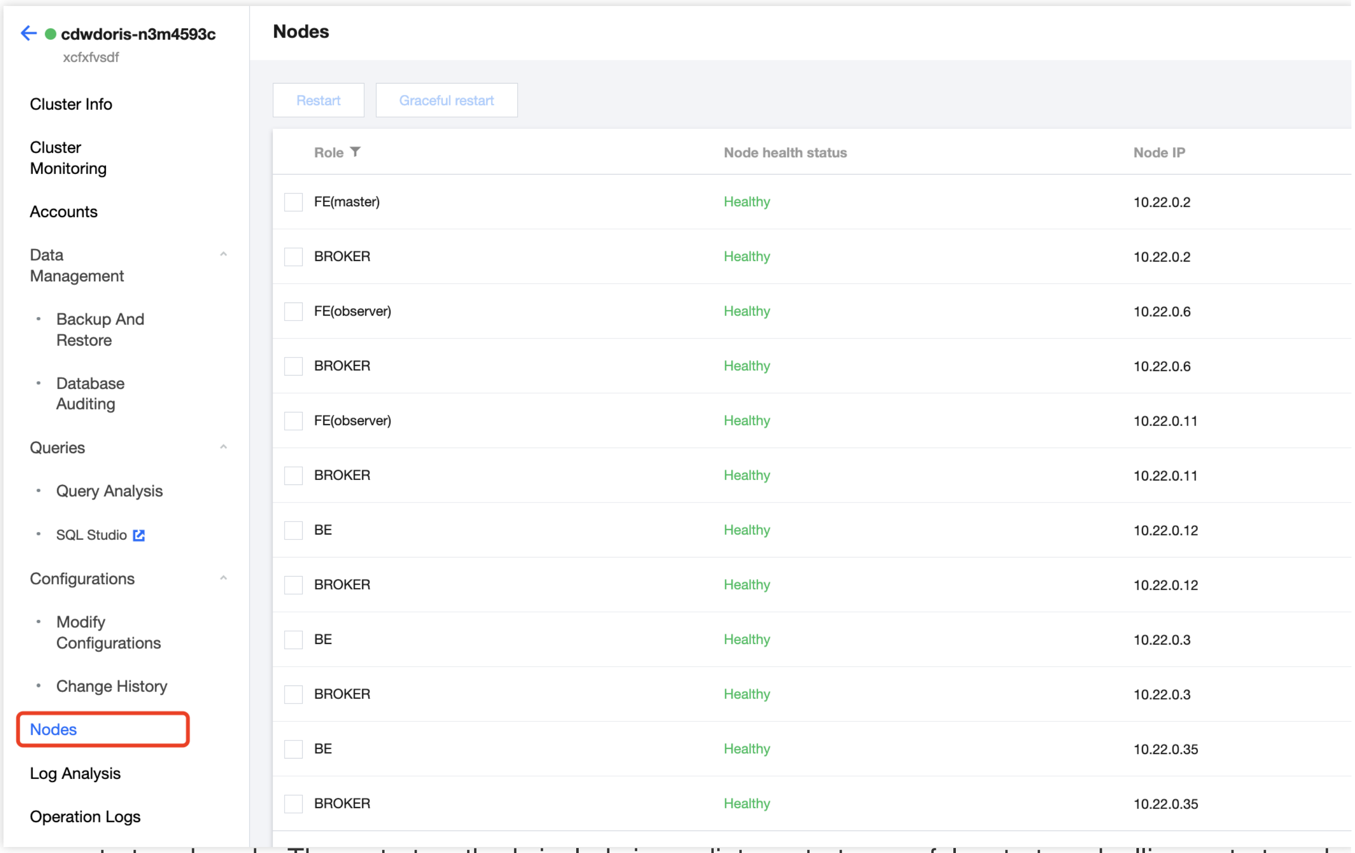


Parameter name	Reference value	Current Configuration Value	Value ran
broker_ipc_port		8000	
client_expire_seconds		300	
XMX	2g	2g	>=2g, <=

2. For the configuration parameters corresponding to Broker, BE, and FE, modifications to the configuration file and hot updates to the parameters are supported. An entire XML configuration file can be modified.

Node Operations

1. The node management page displays the status list of each role in the cluster. The roles include FE, BE, and BROKER. The health status, process status, node IP address, and the most recent restart time (the last time the service was manually restarted on the role management page) are displayed for each role.

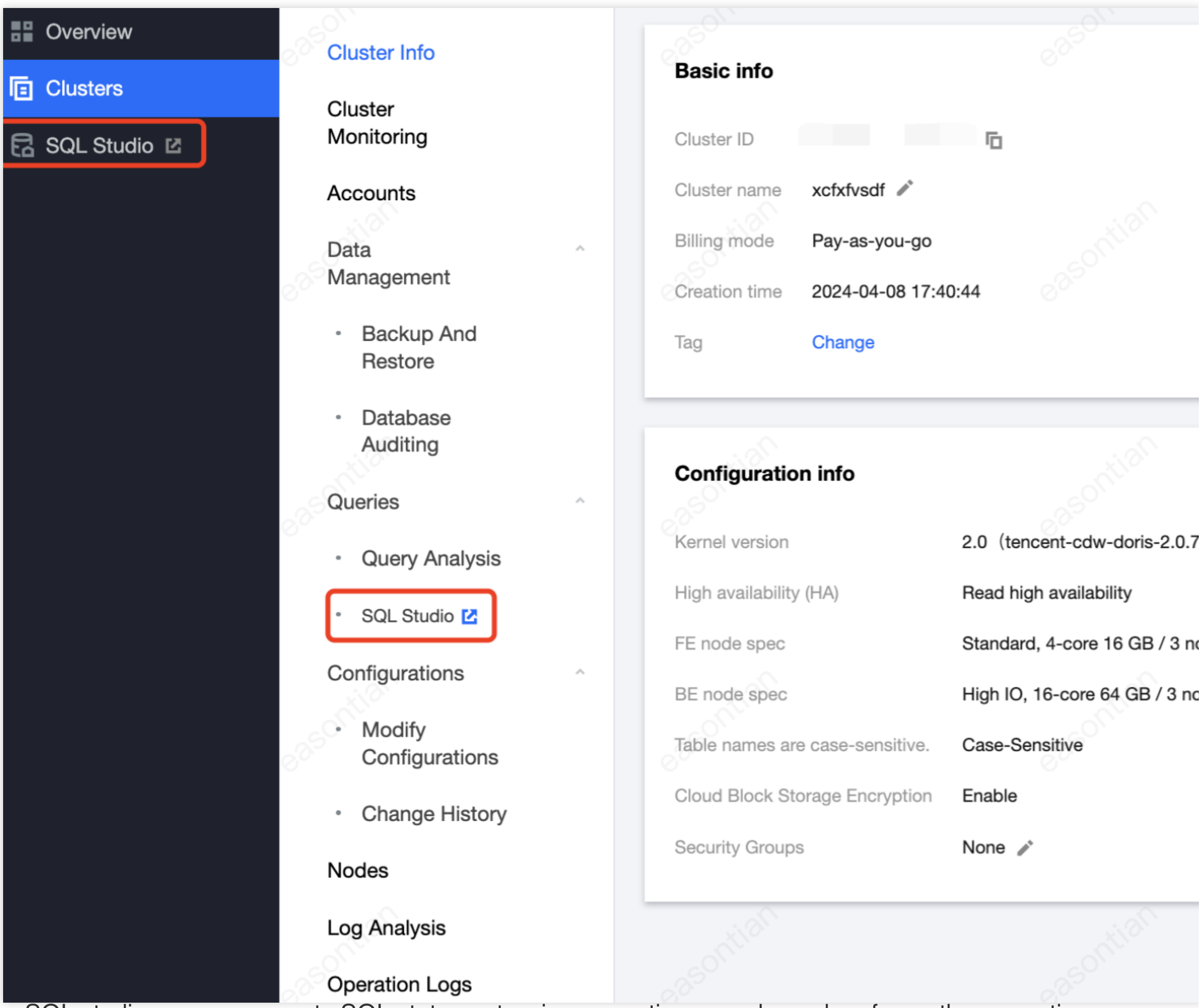


Role	Node health status	Node IP
FE(master)	Healthy	10.22.0.2
BROKER	Healthy	10.22.0.2
FE(observer)	Healthy	10.22.0.6
BROKER	Healthy	10.22.0.6
FE(observer)	Healthy	10.22.0.11
BROKER	Healthy	10.22.0.11
BE	Healthy	10.22.0.12
BROKER	Healthy	10.22.0.12
BE	Healthy	10.22.0.3
BROKER	Healthy	10.22.0.3
BE	Healthy	10.22.0.35
BROKER	Healthy	10.22.0.35

2. You can restart each node. The restart methods include immediate restart, graceful restart, and rolling restart, and batch execution for the same type of nodes is supported.

Executing SQL Commands

1. In the SQL workspace, you can quickly connect to the cluster and use SQL commands to perform a series of operations. For details, see SQL Studio. Click on the console left list to access **SQL Studio**. In the cluster list, select the target **Resource ID**/**Name**. You can also find the entry under Query Management.



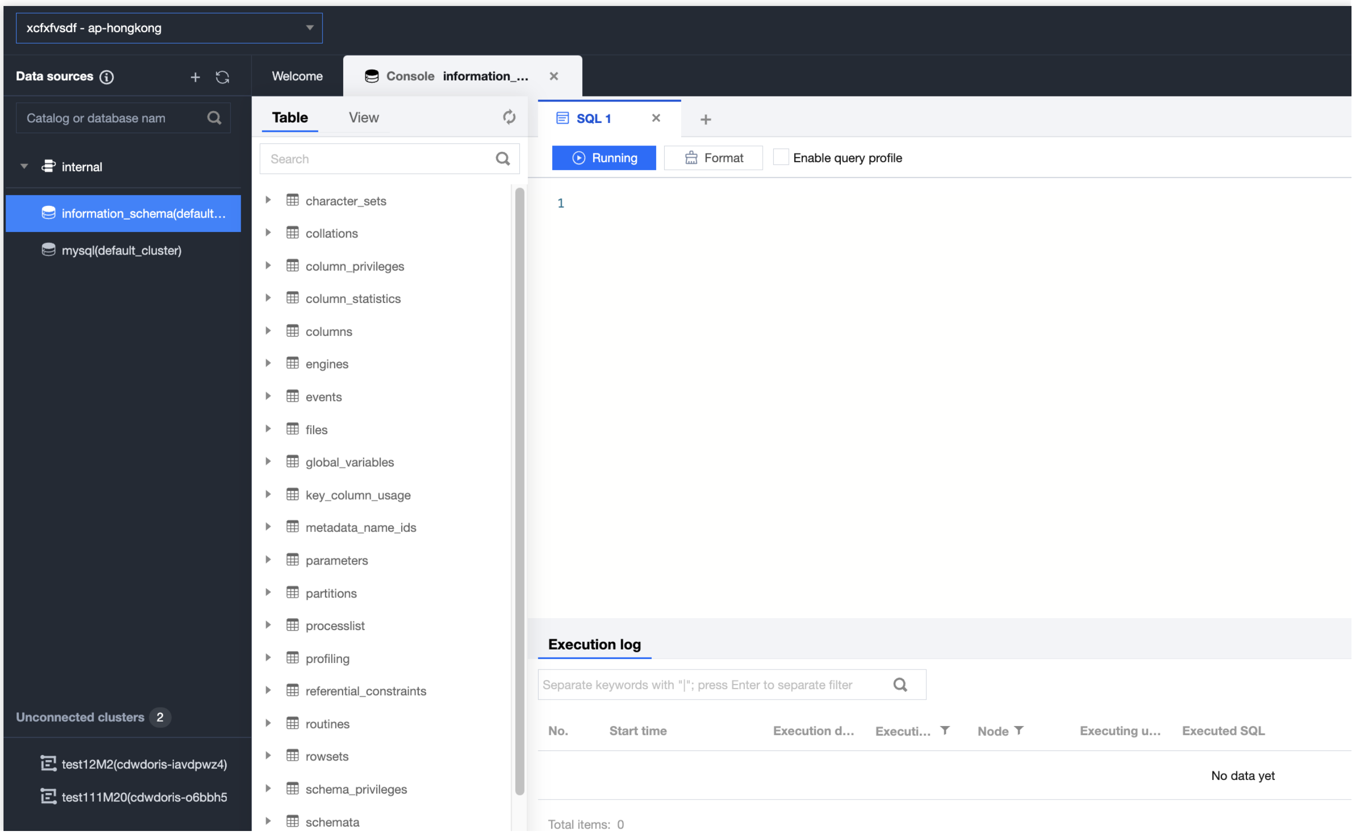
Basic info

Cluster ID	
Cluster name	xcfxfvsdf
Billing mode	Pay-as-you-go
Creation time	2024-04-08 17:40:44
Tag	Change

Configuration info

Kernel version	2.0（tencent-cdw-doris-2.0.7
High availability (HA)	Read high availability
FE node spec	Standard, 4-core 16 GB / 3 nc
BE node spec	High IO, 16-core 64 GB / 3 nc
Table names are case-sensitive.	Case-Sensitive
Cloud Block Storage Encryption	Enable
Security Groups	None

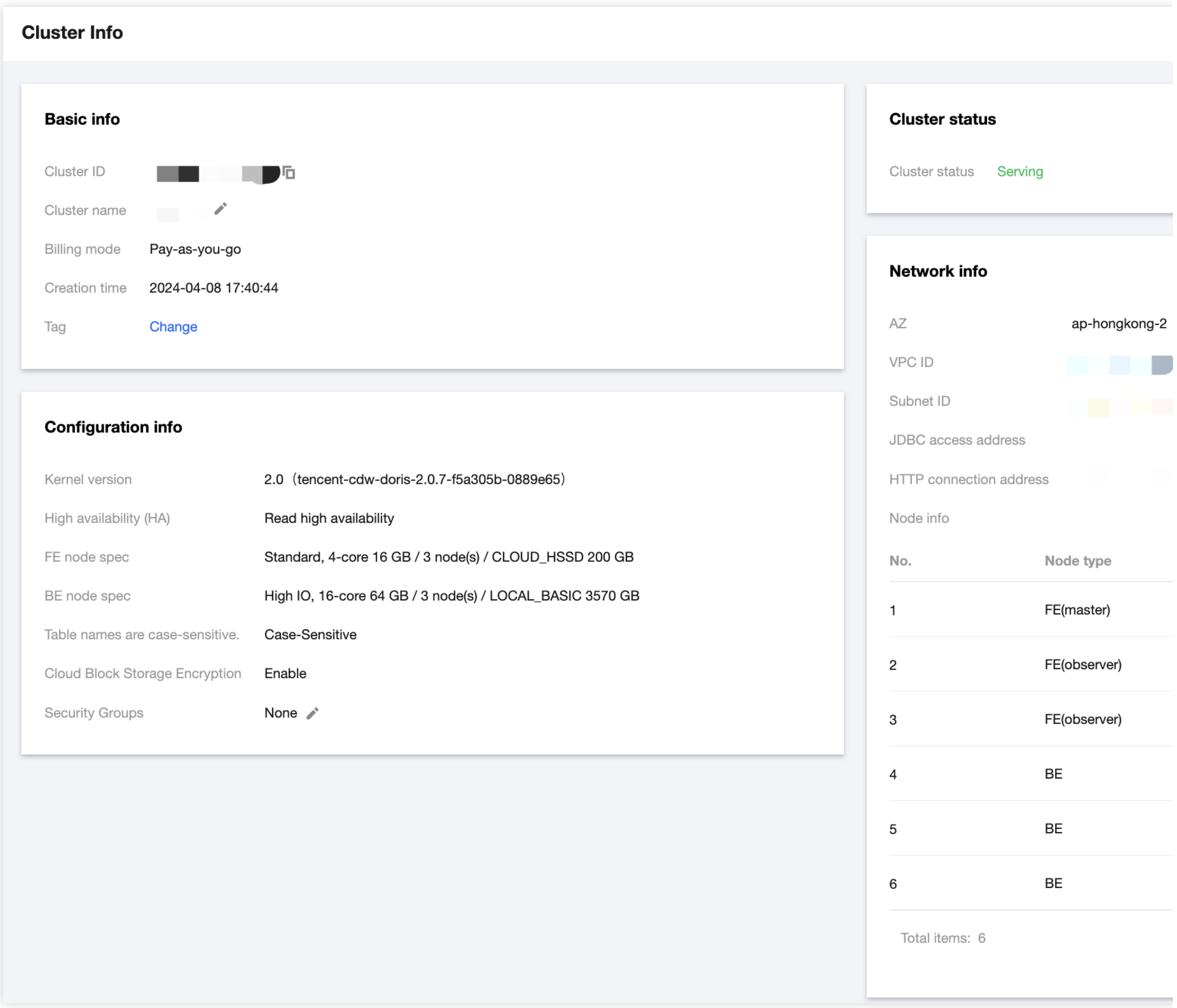
2. In the SQL studio, you can execute SQL statements, view execution records, and perform other operations.


xcfxfvsdf - ap-hongkong
Data sources
Welcome
Console information_...
Catalog or database nam
Table
View
SQL 1
Running
Format
Enable query profile
internal
information_schema(default...
mysql(default_cluster)
Search
character_sets
collations
column_privileges
column_statistics
columns
engines
events
files
global_variables
key_column_usage
metadata_name_ids
parameters
partitions
processlist
profiling
referential_constraints
routines
rowsets
schema_privileges
schemata
Execution log
Separate keywords with "|"; press Enter to separate filter
No.
Start time
Execution d...
Executi...
Node
Executing u...
Executed SQL
No data yet
Total items: 0
Unconnected clusters 2
test12M2(cdwdoris-iavdpwz4)
test111M20(cdwdoris-o6bbh5

Using Tencent Cloud TCHouse-D Through a Client

Last updated：2024-06-27 10:40:06

Except the console, you can use Tencent Cloud TCHouse-D through a client based on JDBC and WebUI. After purchasing and creating a cluster, you can enter the cluster details page from Cluster list by clicking on **Resource ID/name**. The network information section provides two access addresses: JDBC and HTTP, which indicate the two types of clients:
The JDBC connection address is suitable for JDBC clients, such as commonly used MySQL clients.
The HTTP connection address is used to perform various operations on the cluster's REST API through the WebUI.



Cluster Info

Basic info

Cluster ID	[redacted]
Cluster name	[redacted]
Billing mode	Pay-as-you-go
Creation time	2024-04-08 17:40:44
Tag	Change

Configuration info

Kernel version	2.0（tencent-cdw-doris-2.0.7-f5a305b-0889e65）
High availability (HA)	Read high availability
FE node spec	Standard, 4-core 16 GB / 3 node(s) / CLOUD_HSSD 200 GB
BE node spec	High IO, 16-core 64 GB / 3 node(s) / LOCAL_BASIC 3570 GB
Table names are case-sensitive.	Case-Sensitive
Cloud Block Storage Encryption	Enable
Security Groups	None

Cluster status

Cluster status	Serving

Network info

AZ	ap-hongkong-2
VPC ID	[redacted]
Subnet ID	[redacted]
JDBC access address	[redacted]
HTTP connection address	[redacted]
Node info	

No.	Node type
1	FE(master)
2	FE(observer)
3	FE(observer)
4	BE
5	BE
6	BE

Total items: 6

JDBC Clients (MySQL Client and More)

Tencent Cloud TCHouse-D uses the MySQL protocol for communication, and users can connect to the cluster through the MySQL Client or MySQL JDBC. It is recommended that you use the MySQL Client in versions later than 5.1 because the MySQL Client in versions prior to 5.1 does not support usernames longer than 16 characters.

Note:

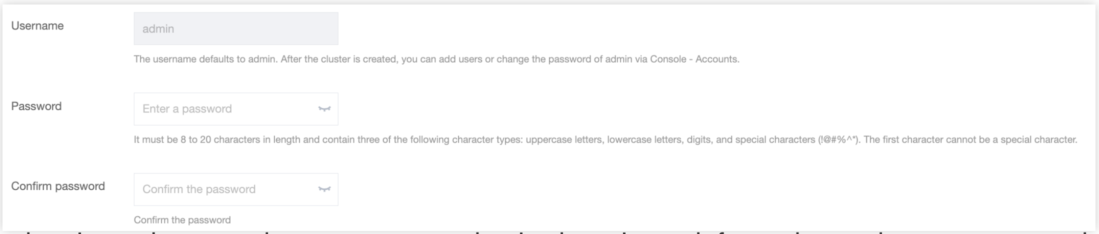
Any client that supports the MySQL JDBC protocol can be used, but there may be situations where the client does not display cluster's return information exactly as it is. Generally, this phenomenon does not cause problems, but operation results may be misjudged because some information is not displayed.

For example, Warning in the return information is not displayed in some clients after Broker Load is executed. Consequently, users may mistakenly think that there is no Warning (that is, no filtered data).

The following describes the basic process of connecting to Tencent Cloud TCHouse-D through MySQL Client.

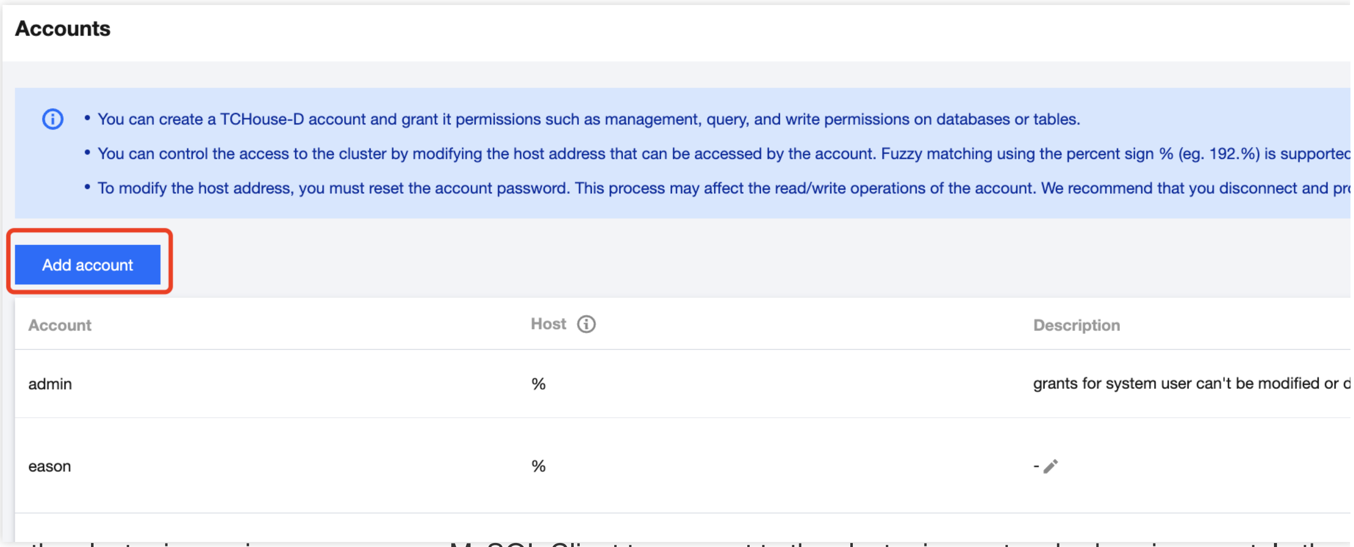
Step 1: Creating a User

1. When you purchase and create a Tencent Cloud TCHouse-D cluster, you need to set the admin user's password.



Username	admin The username defaults to admin. After the cluster is created, you can add users or change the password of admin via Console - Accounts.
Password	Enter a password It must be 8 to 20 characters in length and contain three of the following character types: uppercase letters, lowercase letters, digits, and special characters (!@#%^*). The first character cannot be a special character.
Confirm password	Confirm the password Confirm the password

2. After the cluster is created, you can conveniently view cluster information and carry out operations such as creating new accounts in the console. You can use the admin user to connect to the cluster, or create a new user and use the sub-user to connect to the cluster.



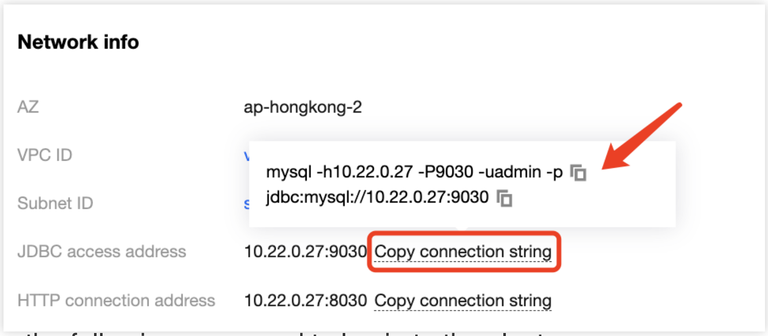
Accounts

- You can create a TCHouse-D account and grant it permissions such as management, query, and write permissions on databases or tables.
- You can control the access to the cluster by modifying the host address that can be accessed by the account. Fuzzy matching using the percent sign % (eg. 192.%) is supportec
- To modify the host address, you must reset the account password. This process may affect the read/write operations of the account. We recommend that you disconnect and pr

Account	Host	Description
admin	%	grants for system user can't be modified or d
eason	%	-

3. When the cluster is running, you can use MySQL Client to connect to the cluster in a networked environment. In the cluster details page, the connection address can be copied with one click. If you want to use a sub-user to connect or

reset the password, you can do so on the **Accounts** page in the console.



Network info

AZ	ap-hongkong-2
VPC ID	
Subnet ID	
JDBC access address	10.22.0.27:9030 Copy connection string
HTTP connection address	10.22.0.27:8030 Copy connection string

```
mysql -h10.22.0.27 -P9030 -uadmin -p
jdbc:mysql://10.22.0.27:9030
```

4. Enter the following command to log in to the cluster:

```
mysql -h FE_HOST -P9030 -uadmin -p
```

Note:

FE_HOST is the IP address of any FE node. 9030 is the query_port configuration in fe.conf.

5. After logging in, you can also modify the admin password using the following command:

```
SET PASSWORD FOR 'admin' = PASSWORD('your_password');
```

6. Create a new user:

```
CREATE USER 'test' IDENTIFIED BY 'test_passwd';
```

7. You can connect to the cluster through the newly created user using the following command:

```
mysql -h FE_HOST -P9030 -utest -ptest_passwd
```

Note:

By default, the newly created ordinary user has no permissions. You can grant permissions **Accounts** page in the console.

Step 2: Creating a Data Table

1. Create a database.

Initially, you can create a database through the admin user: `CREATE DATABASE example_db;` .

Note:
HELP command; can be used to check detailed syntax help for all commands. Example: `HELP CREATE DATABASE;` .
If you do not know the full name of a command, you can use help One field of the command for fuzzy queries. For example, if you type HELP CREATE, you can match commands such as `CREATE DATABASE` , `CREATE TABLE` , and `CREATE USER` .
After the database is created, you can check the database information using `SHOW DATABASES;` .

```
MySQL> SHOW DATABASES;
+--------------------+
| Database           |
+--------------------+
| doris_audit_db__   |
| example_db         |
| information_schema |
+--------------------+
3 rows in set (0.00 sec)
```

information_schema aims to be compatible with the MySQL protocol. Actual information may not be accurate. Therefore, it is recommended that you obtain specific database information by querying the relevant database directly.

2. Grant the account permission.

After example_db is created, you can use the admin account to grant the read/write permissions to the ordinary account, such as test. After the authorization, you can use the test account to operate on the example_db database.

```
GRANT ALL ON example_db TO test;
```

3. Create a table.

First, switch the database:

```
USE example_db;
```

Use the `CREATE TABLE` command to create a table. Tencent Cloud TCHouse-D supports single partition and composite partition to create tables. For details, see Data Partition and Bucketing. The following is an example of creating a table with aggregation model, showcasing both partition types.

Single partition

Create a logical table named table1. The bucketing column is siteid, and the number of buckets is 10. The table's schema is as follows:

siteid: The type is INT (4 bytes), and the default value is 10.

citycode: The type is SMALLINT (2 bytes).
username: The type is VARCHAR, the maximum length is 32, and the default value is an empty character string.
pv: The type is BIGINT (8 bytes), and the default value is 0. This is an index column, the data will be aggregated by default, and the aggregation method is SUM.
The statement for creating the table is as follows:

```
CREATE TABLE table1
(
    siteid INT DEFAULT '10',
    citycode SMALLINT,
    username VARCHAR(32) DEFAULT '',
    pv BIGINT SUM DEFAULT '0'
)
AGGREGATE KEY(siteid,citycode,username)
DISTRIBUTED BY HASH(siteid) BUCKETS 10
PROPERTIES("replication_num" = "1");
```

Composite partition

Create a logical table named table2, with the following schema:

event_day: The type is DATE, with no default value.

siteid: The type is INT (4 bytes), and the default value is 10.

citycode: The type is SMALLINT (2 bytes).

username: The type is VARCHAR, the maximum length is 32, and the default value is an empty character string.

pv: The type is BIGINT (8 bytes), and the default value is 0. This is an index column, the data will be aggregated by default, and the aggregation method is SUM.

Use the event_day column as the partition column and create 3 partitions, including p201706, p201707, and p201708. Each partition uses siteid for hash bucket, and the number of buckets is 10.

p201706: The range is [minimum value, 2017-07-01).

p201707: The range is [2017-07-01, 2017-08-01).

p201708: The range is [2017-08-01, 2017-09-01).

The statement for creating the table is as follows:

```
CREATE TABLE table2
(
    event_day DATE,
    siteid INT DEFAULT '10',
    citycode SMALLINT,
    username VARCHAR(32) DEFAULT '',
    pv BIGINT SUM DEFAULT '0'
)
AGGREGATE KEY(event_day, siteid, citycode, username)
PARTITION BY RANGE(event_day)
(
    PARTITION p201706 VALUES LESS THAN ('2017-07-01'),
    PARTITION p201707 VALUES LESS THAN ('2017-08-01'),
    PARTITION p201708 VALUES LESS THAN ('2017-09-01')
)
DISTRIBUTED BY HASH(siteid) BUCKETS 10
PROPERTIES("replication_num" = "1");
```

After the table is created, you can check the information of the table in example_db:

```
MySQL> SHOW TABLES;
+----------------------+
| Tables_in_example_db |
+------------------------+
| table1                 |
| table2                 |
+------------------------+
2 rows in set (0.01 sec)

MySQL> DESC table1;
+----------+-------------+------+-------+---------+-------+
| Field    | Type        | Null | Key   | Default | Extra |
+----------+-------------+------+-------+---------+-------+
| siteid   | int(11)     | Yes  | true  | 10      |       |
| citycode | smallint(6) | Yes  | true  | N/A     |       |
| username | varchar(32) | Yes  | true  |         |       |
| pv       | bigint(20)  | Yes  | false | 0       | SUM   |
+----------+-------------+------+-------+---------+-------+
4 rows in set (0.00 sec)

MySQL> DESC table2;
+-----------+-------------+------+-------+---------+-------+
| Field     | Type        | Null | Key   | Default | Extra |
+-----------+-------------+------+-------+---------+-------+
| event_day | date        | Yes  | true  | N/A     |       |
| siteid    | int(11)     | Yes  | true  | 10      |       |
| citycode  | smallint(6) | Yes  | true  | N/A     |       |
| username  | varchar(32) | Yes  | true  |         |       |
| pv        | bigint(20)  | Yes  | false | 0       | SUM   |
+-----------+-------------+------+-------+---------+-------+
5 rows in set (0.00 sec)
```

Note:

In the above tables, tables created through setting replication_num are single-copy tables. For Tencent Cloud TCHouse-D, it is recommended that users use the default 3-copy setting to ensure high availability.

Step 3: Importing Data

Tencent Cloud TCHouse-D supports multiple data import methods. For specific details, see Importing Data. Examples of streaming import and Broker import are provided here.

Streaming Import

Streaming import transmits data to Tencent Cloud TCHouse-D using the HTTP protocol, allowing for the direct import of local data without dependency on other systems or components. For detailed syntax help, see `HELP STREAM LOAD;` .

Example 1: "table1_20170707" is used as Label, and the local file table1_data is imported in table1.

```
curl --location-trusted -u test:test_passwd -H "label:table1_20170707" -H "column_s
```

Note:

FE_HOST is the IP address of any FE's node location, and 8030 is http_port in fe.conf.

You can use any BE's IP address and webserver_port in be.conf for importing. Example: BE_HOST:8040.

For the local file table1_data, the comma (,) is used as a separator between data, as follows:

```
1,1,jim,2
2,1,grace,2
3,2,tom,2
4,3,bush,3
5,3,helen,3
```

Example 2: table2_20170707 is used as Label, and the local file table2_data is imported to table2.

```
curl --location-trusted -u test:test -H "label:table2_20170707" -H "column_separato
http://127.0.0.1:8030/api/example_db/table2/_stream_load
```

For the local file `table2_data` , `|` is used as the separator between data, as follows:

```
2017-07-03|1|1|jim|2
2017-07-05|2|1|grace|2
2017-07-12|3|2|tom|2
2017-07-15|4|3|bush|3
2017-07-12|5|3|helen|3
```

Note:

For streaming import, it is recommended that the file size be within 10 GB. Files that exceed this size could cause import failure and increase the cost of retries.

Each batch of data import requires a unique Label. The Label should ideally be a character string related to the batch of data, facilitating reading and management. Tencent Cloud TCHouse-D guarantees that within a database, the same batch of data can only be imported only once based on the Label. Labels of failed tasks can be reused.
The streaming import command is executed in a synchronous manner. If success is returned for the command, the data has been imported. If failure is returned for the command, the data has not been imported.

Broker Import

Broker Import uses a deployed Broker process to read and import data from external storage. For more help, see `HELP BROKER LOAD;` .
Example: table1_20170708 is used as Label, to import files from HDFS into table1.

```
LOAD LABEL table1_20170708
(
 DATA INFILE("hdfs://your.namenode.host:port/dir/table1_data")
 INTO TABLE table1
)
WITH BROKER hdfs
(
 "username"="hdfs_user",
 "password"="hdfs_password"
)
PROPERTIES
```

```
(
 "timeout"="3600",
 "max_filter_ratio"="0.1"
);
```

The Broker import command is executed in an asynchronous manner. The successful execution of the above command only indicates the successful submission of tasks. You can check whether the import is successful through `SHOW LOAD;` . Example:

```
SHOW LOAD WHERE LABEL = "table1_20170708";
```

In the returned result, if the `State` field is FINISHED, the import is successful. For more information about SHOW LOAD, see `HELP SHOW LOAD;` .
Asynchronous import tasks can be cancelled before they end: `CANCEL LOAD WHERE LABEL = "table1_20170708";` .

Step 4: Querying Data

1. Simple Query:

```
MySQL> SELECT * FROM table1 LIMIT 3;
+---------+----------+----------+------+
| siteid | citycode | username | pv   |
+--------+----------+----------+------+
|      2 |        1 | 'grace'  |    2 |
|      5 |        3 | 'helen'  |    3 |
|      3 |        2 | 'tom'    |    2 |
+--------+----------+----------+------+
3 rows in set (0.01 sec)

MySQL> SELECT * FROM table1 ORDER BY citycode;
+--------+----------+----------+------+
| siteid | citycode | username | pv   |
+--------+----------+----------+------+
|      2 |        1 | 'grace'  |    2 |
|      1 |        1 | 'jim'    |    2 |
|      3 |        2 | 'tom'    |    2 |
|      4 |        3 | 'bush'   |    3 |
|      5 |        3 | 'helen'  |    3 |
+--------+----------+----------+------+
5 rows in set (0.01 sec)
```

2. Join Query:

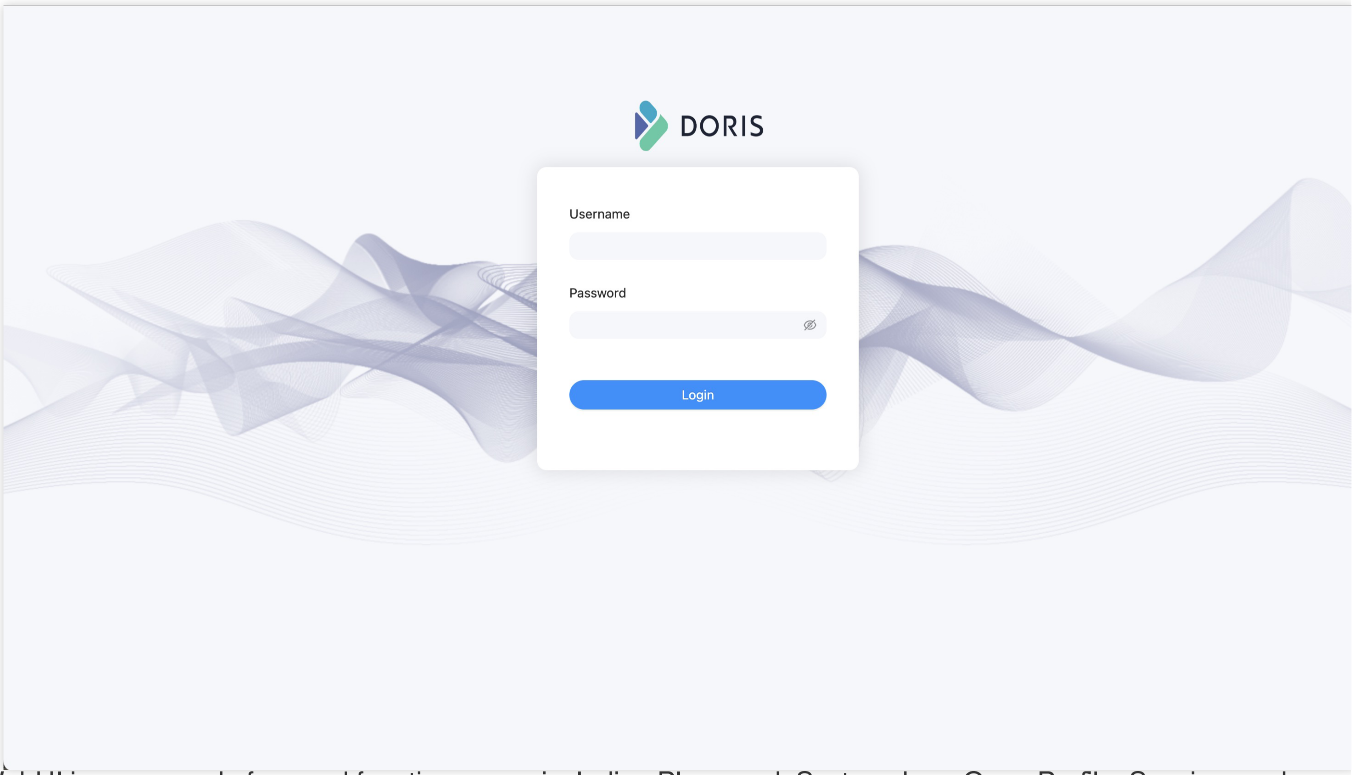
3. Subquery:

```
MySQL> SELECT SUM(table1.pv) FROM table1 JOIN table2 WHERE table1.siteid = table2.s
+--------------------+
| sum(`table1`.`pv`) |
+--------------------+
|                 12 |
+--------------------+
1 row in set (0.20 sec)
```

```
MySQL> SELECT SUM(pv) FROM table2 WHERE siteid IN (SELECT siteid FROM table1 WHERE
+-----------+
| sum(`pv`) |
+-----------+
|         8 |
+-----------+
1 row in set (0.13 sec)
```

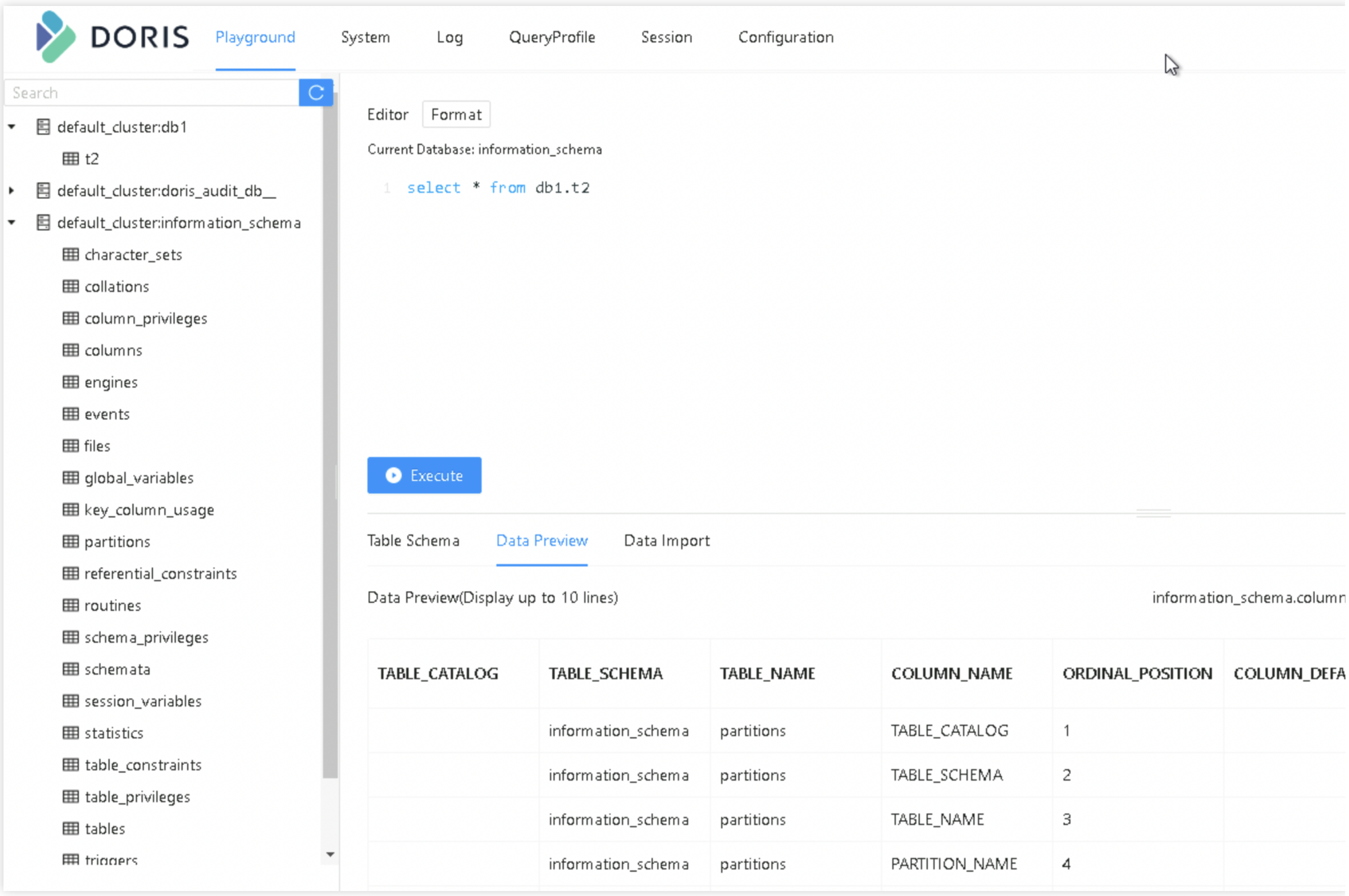
WebUI

The WebUI is based on JDBC and Rest API capabilities, allowing you to perform data operations and manage clusters among other features. To use the WebUI, enter `https:// fe_ip:8030` in a networked environment. The following page is displayed, it has started successfully.



The WebUI is composed of several function pages including Playgroud, System, Log, QueryProfile, Session, and Configuration.

Playground provides an SQL query editor for executing various SQL commands. The left side displays all databases and their tables in a tree-like structure. You can double-click on a table name to check the metadata and sample data of the table. After you click on `Data Import` , data can be uploaded from local storage.



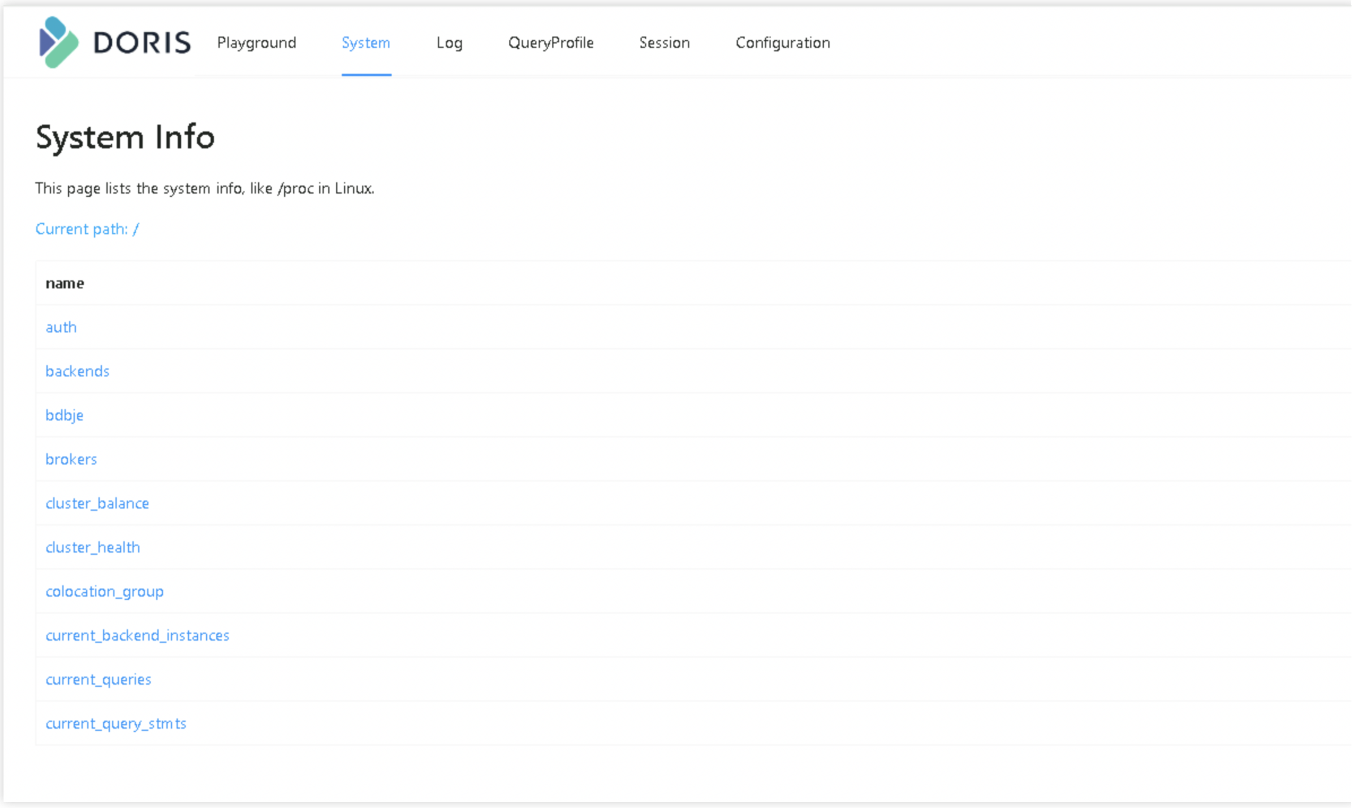
Current Database: information_schema

```
select * from db1.t2
```

Data Preview(Display up to 10 lines) information_schema.column

TABLE_CATALOG	TABLE_SCHEMA	TABLE_NAME	COLUMN_NAME	ORDINAL_POSITION	COLUMN_DEFA
	information_schema	partitions	TABLE_CATALOG	1	
	information_schema	partitions	TABLE_SCHEMA	2	
	information_schema	partitions	TABLE_NAME	3	
	information_schema	partitions	PARTITION_NAME	4	

The System page provides various system information about the cluster.



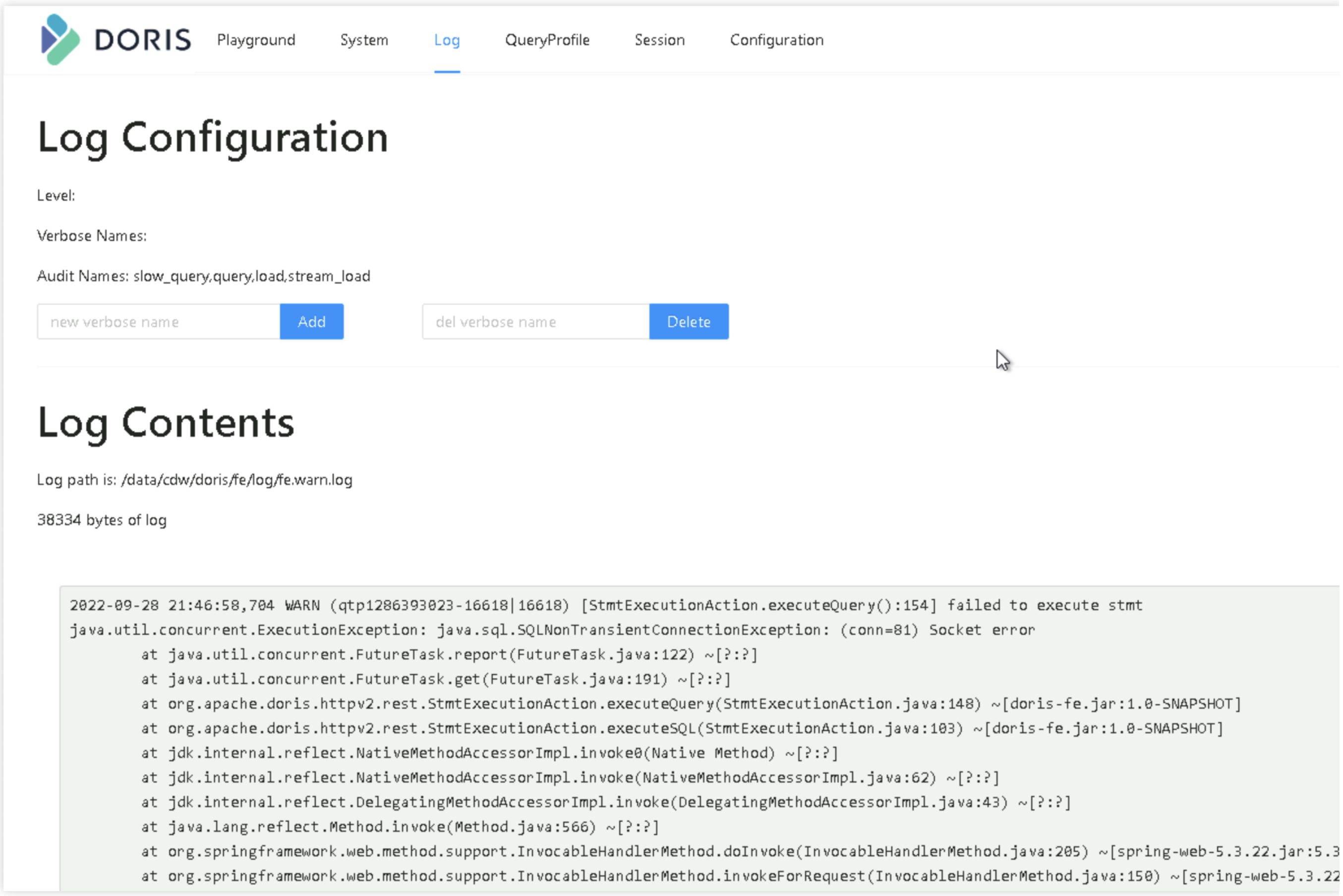
System Info

This page lists the system info, like /proc in Linux.

Current path: /

name
auth
backends
bdbje
brokers
cluster_balance
cluster_health
colocation_group
current_backend_instances
current_queries
current_query_stmts

The Log page provides features for displaying and managing the FE logs.



Log Configuration

Level:

Verbose Names:

Audit Names: slow_query,query,load,stream_load

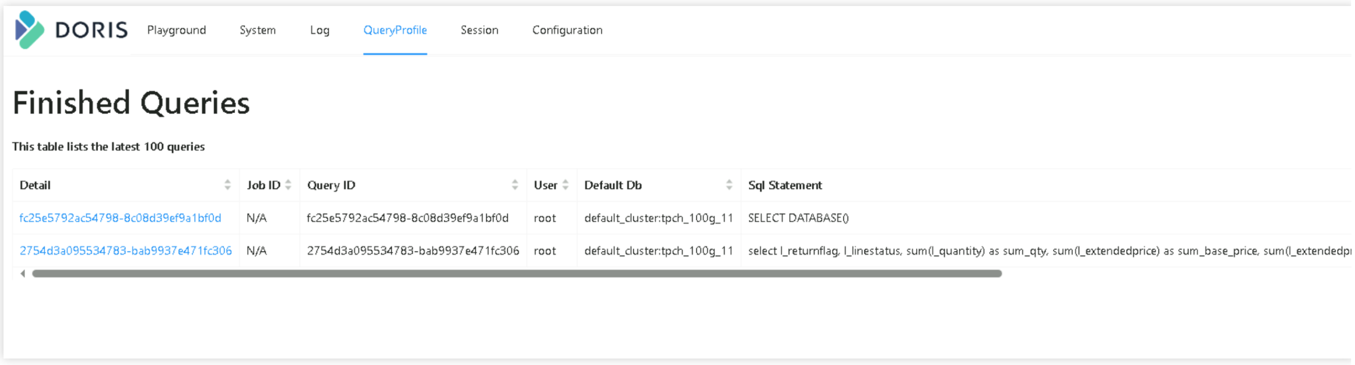
Log Contents

Log path is: /data/cdw/doris/fe/log/fe.warn.log

38334 bytes of log

```
2022-09-28 21:46:58,704 WARN (qtp1286393023-16618|16618) [StmtExecutionAction.executeQuery():154] failed to execute stmt
java.util.concurrent.ExecutionException: java.sql.SQLNonTransientConnectionException: (conn=81) Socket error
        at java.util.concurrent.FutureTask.report(FutureTask.java:122) ~[?:?]
        at java.util.concurrent.FutureTask.get(FutureTask.java:191) ~[?:?]
        at org.apache.doris.httpv2.rest.StmtExecutionAction.executeQuery(StmtExecutionAction.java:148) ~[doris-fe.jar:1.0-SNAPSHOT]
        at org.apache.doris.httpv2.rest.StmtExecutionAction.executeSQL(StmtExecutionAction.java:103) ~[doris-fe.jar:1.0-SNAPSHOT]
        at jdk.internal.reflect.NativeMethodAccessorImpl.invoke0(Native Method) ~[?:?]
        at jdk.internal.reflect.NativeMethodAccessorImpl.invoke(NativeMethodAccessorImpl.java:62) ~[?:?]
        at jdk.internal.reflect.DelegatingMethodAccessorImpl.invoke(DelegatingMethodAccessorImpl.java:43) ~[?:?]
        at java.lang.reflect.Method.invoke(Method.java:566) ~[?:?]
        at org.springframework.web.method.support.InvocableHandlerMethod.doInvoke(InvocableHandlerMethod.java:205) ~[spring-web-5.3.22.jar:5.3
        at org.springframework.web.method.support.InvocableHandlerMethod.invokeForRequest(InvocableHandlerMethod.java:150) ~[spring-web-5.3.22
```

The QueryProfile page displays SQL recorded in the profile, where you can check the overall SQL as well as the execution plan and runtime data for each instance.

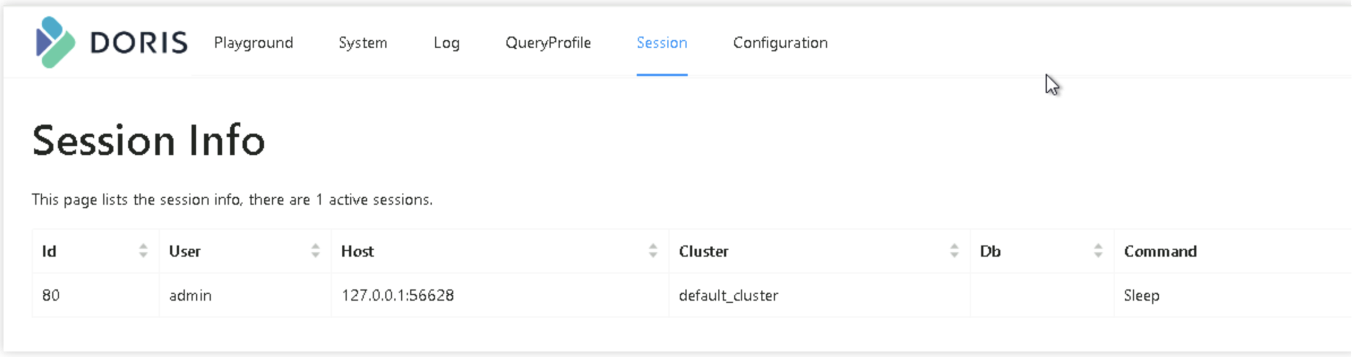


Finished Queries

This table lists the latest 100 queries

Detail	Job ID	Query ID	User	Default Db	Sql Statement
fc25e5792ac54798-8c08d39ef9a1bf0d	N/A	fc25e5792ac54798-8c08d39ef9a1bf0d	root	default_cluster:tpch_100g_11	SELECT DATABASE()
2754d3a095534783-bab9937e471fc306	N/A	2754d3a095534783-bab9937e471fc306	root	default_cluster:tpch_100g_11	select l_returnflag, l_linestatus, sum(l_quantity) as sum_qty, sum(l_extendedprice) as sum_base_price, sum(l_extendedpr

The Session page displays information about all active sessions.

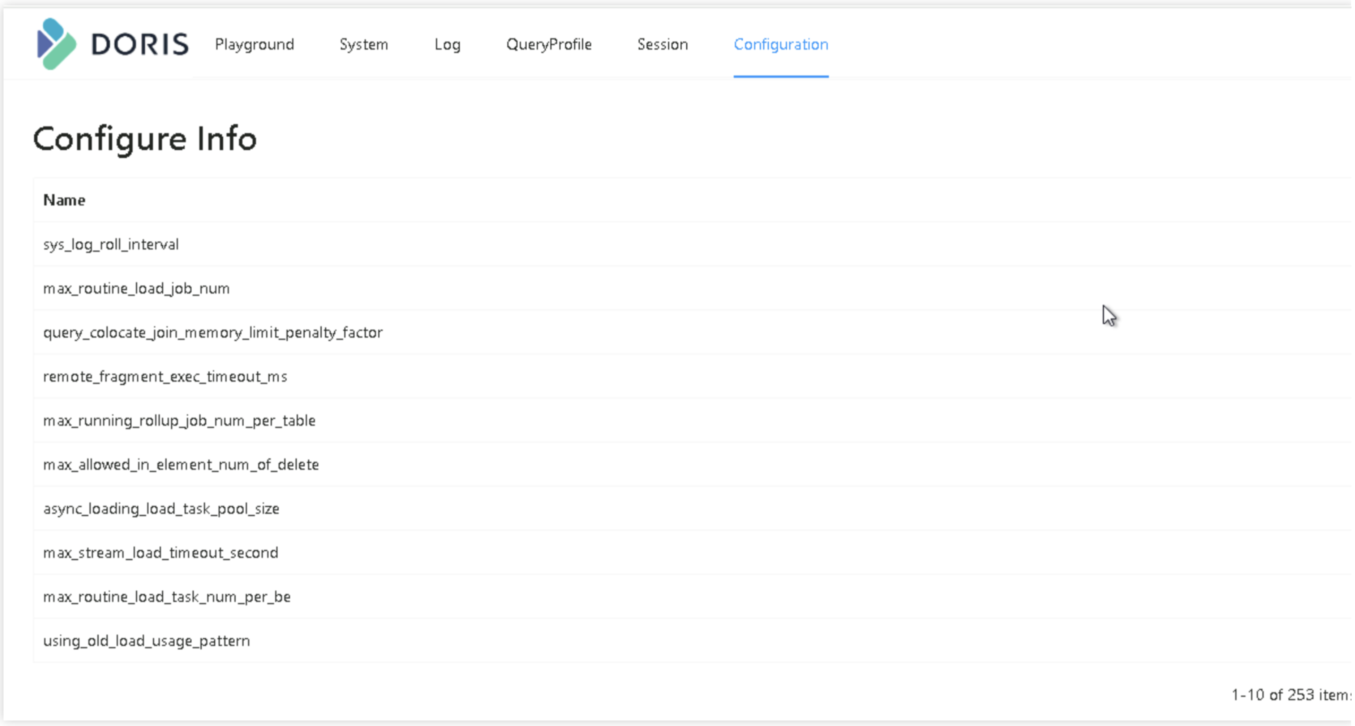


Session Info

This page lists the session info, there are 1 active sessions.

Id	User	Host	Cluster	Db	Command
80	admin	127.0.0.1:56628	default_cluster		Sleep

The Configuration page provides all configuration information of the FE, offering a filtering feature in the configuration value column.



Configure Info

Name
sys_log_roll_interval
max_routine_load_job_num
query_colocate_join_memory_limit_penalty_factor
remote_fragment_exec_timeout_ms
max_running_rollup_job_num_per_table
max_allowed_in_element_num_of_delete
async_loading_load_task_pool_size
max_stream_load_timeout_second
max_routine_load_task_num_per_be
using_old_load_usage_pattern

1-10 of 253 item: