Tencent Cloud TCHouse-P

Operation Guide

Service Notice

This document provides an overview of the as-is details of Tencent Cloud's products and services in their entirety or part. The descriptions of certain products and services may be subject to adjustments from time to time.

The commercial contract concluded by you and Tencent Cloud will provide the specific types of Tencent Cloud products and services you purchase and the service standards. Unless otherwise agreed upon by both parties, Tencent Cloud does not make any explicit or implied commitments or warranties regarding the content of this document.

Contact Us

We are committed to providing personalized pre-sales consultation and technical after-sale support. Don't hesitate to contact us at 4009100100 or 95716 for any inquiries or concerns.

Contents

Operation Guide
Cluster Management
Creating cluster

Last updated: 2024-08-22 15:35:12

Operation scenarios

This document describes how to create a cluster in the Tencent Cloud TCHouse-P console.

Operation step

1. Go to Tencent Cloud TCHouse-P and select Cluster List in the console.
2. Click **Create Cluster** to enter the purchase page and fill in the required parameters accordingly.

- **Cluster Name:** Used to distinguish between clusters.
- **Billing Mode:** Monthly subscription and pay-as-you-go billing are supported.
- **Region:** Currently, Beijing, Chengdu, Chongqing, Guangzhou, Shanghai, Shanghai Finance, Singapore, and Virginia regions are supported.
- **Availability Zone:** Different AZs are not interconnected.
- **Network Information:** Select the VPC network and subnet information you wish to connect to the Tencent Cloud TCHouse-P. You can query or plan in VPC. If you need other subnets or public network access, refer to Applying for Public IP and Creating IP Allowlist.
- **Password:** The default username is dbadmin. This is the login password required to connect to the Tencent Cloud TCHouse-P cluster.
- **Node Types:** CN node, DN node.
- **Node Count:** Supports 2 to 50 nodes, with a general requirement of at least 2 compute nodes, up to a maximum of 50 nodes.

- **Database Port:** 9000, currently cannot be modified.

基础配置

集群名称 *	请输入集群名称

支持中文、字母、数字、-、_，6-36个字符

付费模式 *　包年包月　按量计费

地域 *
———中国———　　　·东南亚地区·　—北美—
北京　成都　重庆　广州　上海　上海金融　新加坡　弗吉尼亚

可用区 *　北京六区　北京七区

网络信息
网络 *　…1 | 子网删除测试vp ▾　↻ 新建私有网络 ↗
子网 *　…: | 北京七区初½ ▾　↻ 新建子网 ↗
共 253 个子网 IP，剩 236 个可用

网络/子网选项仅展示20条，更多选项可通过名称关键字搜索获取

账户信息
密码 *　请输入密码　∅
以字母、数字开头，包含小写字母、大写字母、数字、特殊字符(!@#%^)中的三种，8-16位字符
确认密码 *　请输入密码　∅

默认用户名为dbadmin，集群创建后，您可以通过「账户管理」修改密码

集群配置

CN 节点配置 *　标准型　高性能型　大数据型
计算规格 *　4核16G ▾
存储规格 *　增强型SSD云硬盘 ▾　－　200　＋　GB
单节点最小支持 200GB，最大支持 32000GB，步长为100
节点数量 *　－　2　＋
节点数量大于等于2台，小于等于1000台，库存节点1576，库存不足时请更换规格或 联系我们 ↗ 协调资源

DN 节点配置 *　标准型　高性能型　大数据型
计算规格 *　4核16G ▾
存储规格 *　增强型SSD云硬盘 ▾　－　200　＋　GB
单节点最小支持 200GB，最大支持 320000GB，步长为100
节点数量 *　－　2　＋
节点数量大于等于2台，小于等于1000台，库存节点1999，库存不足时请更换规格或 联系我们 ↗ 协调资源
数据复制方式 *　● 强同步（可退化）　○ 异步　○ 强同步
仅存活 1 个节点时，实例将自动退化为异步复制

字符集 *　● UTF8　○ LATIN1　○ EUC_CN　○ SQL_ASCII
若字符集设置不当会导致数据库导入发生错误

其他配置

标签信息　标签键　标签值　✕
＋ 添加　⊙ 键值粘贴板
标签用于从不同维度对资源分类管理，标签上限为5个，键值不完整将自动丢弃，如需了解更多，请前往 标签产品文档 ↗

3. After configuration, click **Purchase Now**.

4. Return to the cluster list. Once the cluster status is **Running**, you can start using the cluster.

Cluster Information

Last updated：2024-08-22 15:35:41

This article introduces how to modify the basic information, billing model, and view the performance monitoring of the Tencent Cloud TCHouse-P through the console.

Basic information

Log in to **Tencent Cloud TCHouse-P** and click **Cluster ID/Name** to go to the **Basic Configuration** page, where you can set the cluster name, network, and manage user passwords.

> ⚠ **Note**
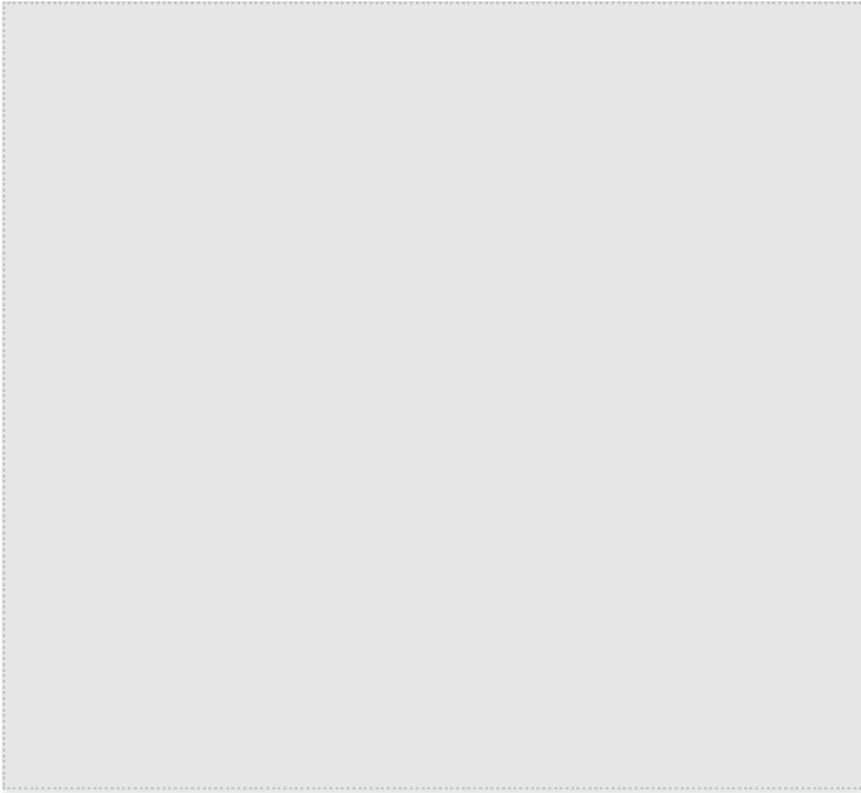> Changing the network address will change the cluster access connection, so you need to change the call address simultaneously.
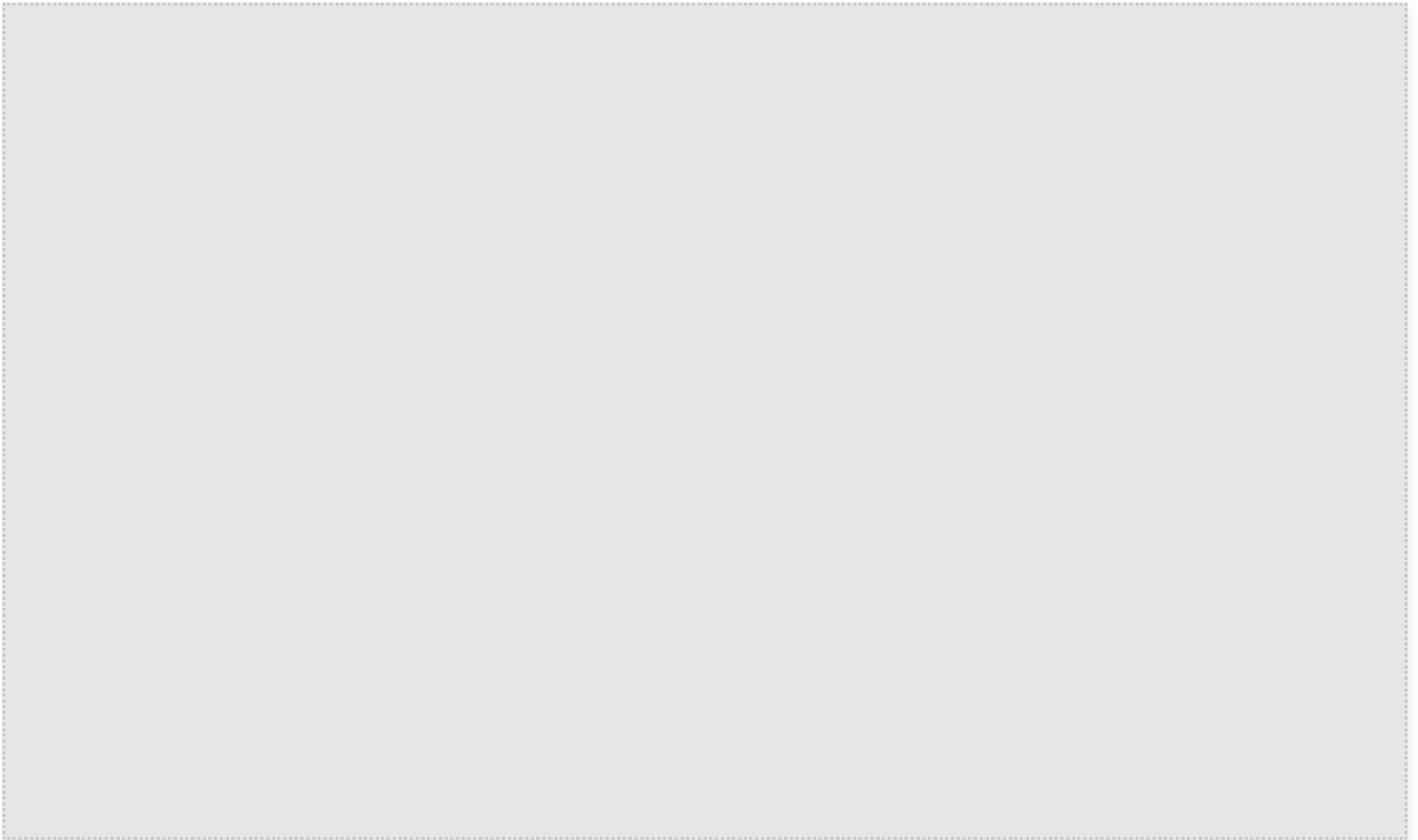

Billing Information

On the **Basic Configuration** page, you can switch from pay-as-you-go to monthly subscription billing mode for the cluster and enable **auto-renewal** as needed.

Performance Monitoring

On the **Performance Monitoring** page, you can view various metrics in secondary node and cluster dimensions.

Scaling Cluster

Last updated：2024-08-22 15:36:13

Operation scenarios

When computing or storage resources of a cluster reach a bottleneck, you can increase system resources by scaling.

Operation step

1. log in to Tencent Cloud TCHouse-P , in the **Operation** column of the cluster list, click **Manage** to enter the cluster details page.
2. Click the top-right **Scale-out** to enter the Expansion Page and select the information as needed.

> ⚠ **Note**
> The number of nodes entered is that after scaling, which must be greater than the current quantity by two or more. The value of the total capacity changes as the number of nodes changes.

3. After determining the scaling quantity, click **Confirm** to start the scaling process.
4. Return to the cluster list, and the cluster status during scaling will be **Scaling out**. The scaling time depends on the existing data volume in the cluster. After a successful scale, the cluster status will return to **Running**.

Managing IP Allowlist and Blocklist

Last updated: 2024-08-22 15:36:56

Operation scenarios

Other than accessing the Tencent Cloud TCHouse-P cluster through CVM instances in the specified subnets, if you need to access the Tencent Cloud TCHouse-P cluster through CVM instances in other subnets or servers on the public network, you need to add these IP addresses to the allowlist.

An IP range can be specified when an IP is added to the allowlist. Add one or more IPs in the IP range to the blocklist if you want to block the access to CDWPG clusters from them.

> ⚠ **Note**
> The priority of the blocklist is higher than the allowlist.

Prerequisites

Before using the IP allowlist, you need to ensure that a public IP for the cluster has been applied for. Otherwise, even if their addresses are added to the allowlist, you will not be able to access the Tencent Cloud TCHouse-P cluster from servers other than CVM instances in the specified subnets.

Operation step

Managing IP allowlist

1. In the cluster list, click **Manage** to enter the cluster details page.
2. Select **Configure** > **Access allowlist**.
3. Click **Create Allowlist** and enter the IP, username, database name, and CIDR address as required.

4. Click **OK** to add the IP range specified by the CIDR block to the allowlist. Once added successfully, hosts in that address range or network segment can access the Tencent Cloud TCHouse-P cluster.

> ⓘ **Note**
> When an IP allowlist is no longer needed, select the corresponding address in the allowlist and click **Delete** to remove it.

Managing IP blocklist

1. Select **Configure** > **Access blocklist**.
2. Click **Create Blocklist** and enter the IP, username, database name, and CIDR address as required.
3. Click **OK** to add the IP range specified by the CIDR block to the blocklist. Once added successfully, hosts in that address range or network segment will not be able to access the Tencent Cloud TCHouse-P cluster.

> **Note**
> To remove the access restriction for an IP or IP range, select the corresponding address in the blocklist and click **Delete**.

Public network activation

Last updated：2024−08−22 15:37:22

Create public network CLB

1. On the **Instance Management** page, click **Create**.

2. Select **Billing Model** and **Region**. Confirm the network type is **"Public Network"**. Choose **Network** and **Bandwidth Cap**.
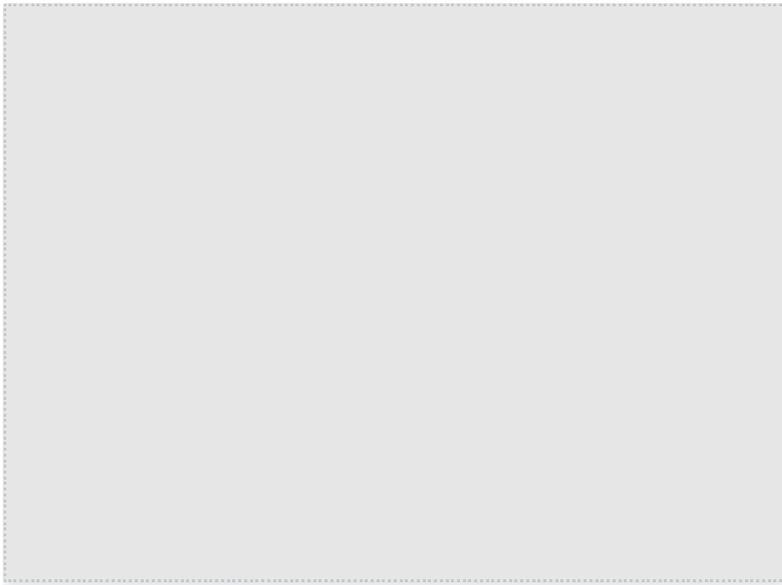
Create Listener

Enter the CLB control panel and select "Listener Management"
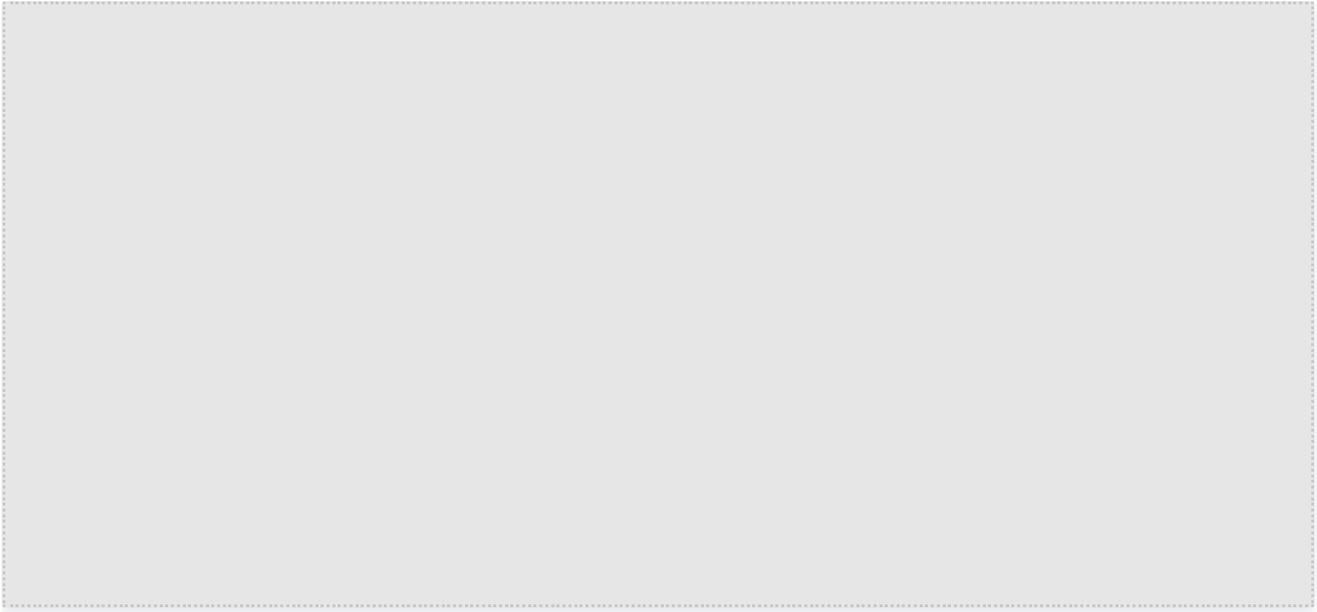
Click "Create", fill in the listening port, and click "Next"
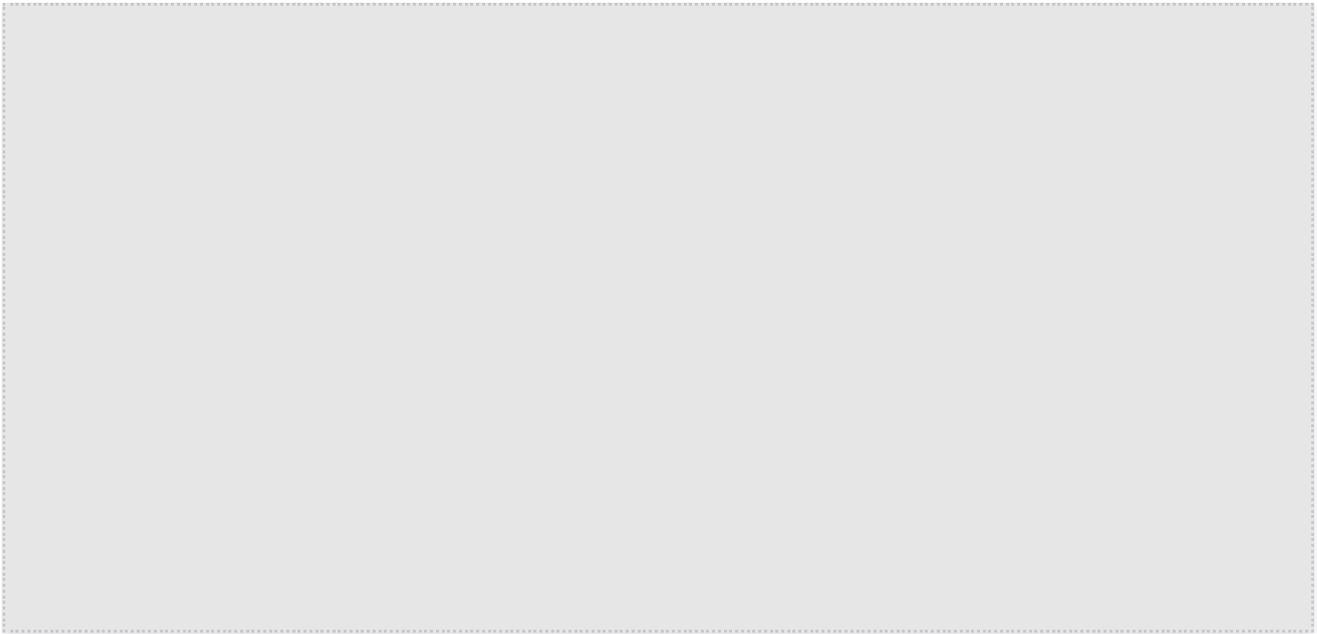
Set "Health Check", fill in the "Check Port", and click "Next"
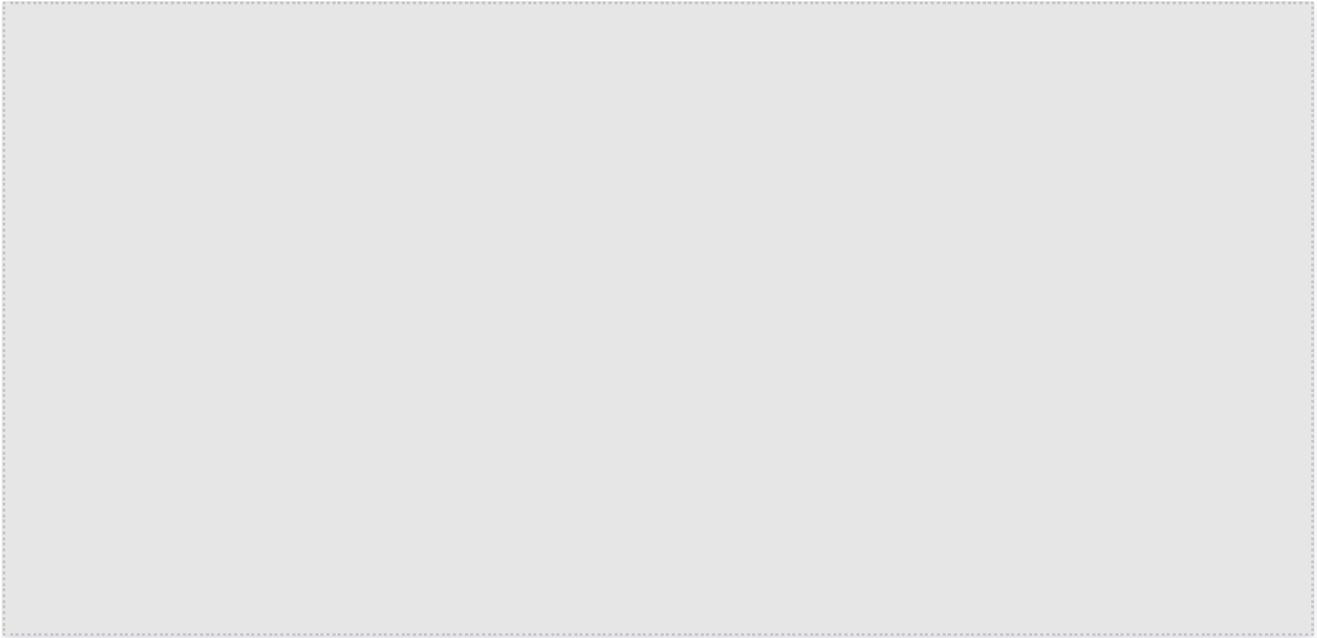
Click "Submit" to complete the listener creation
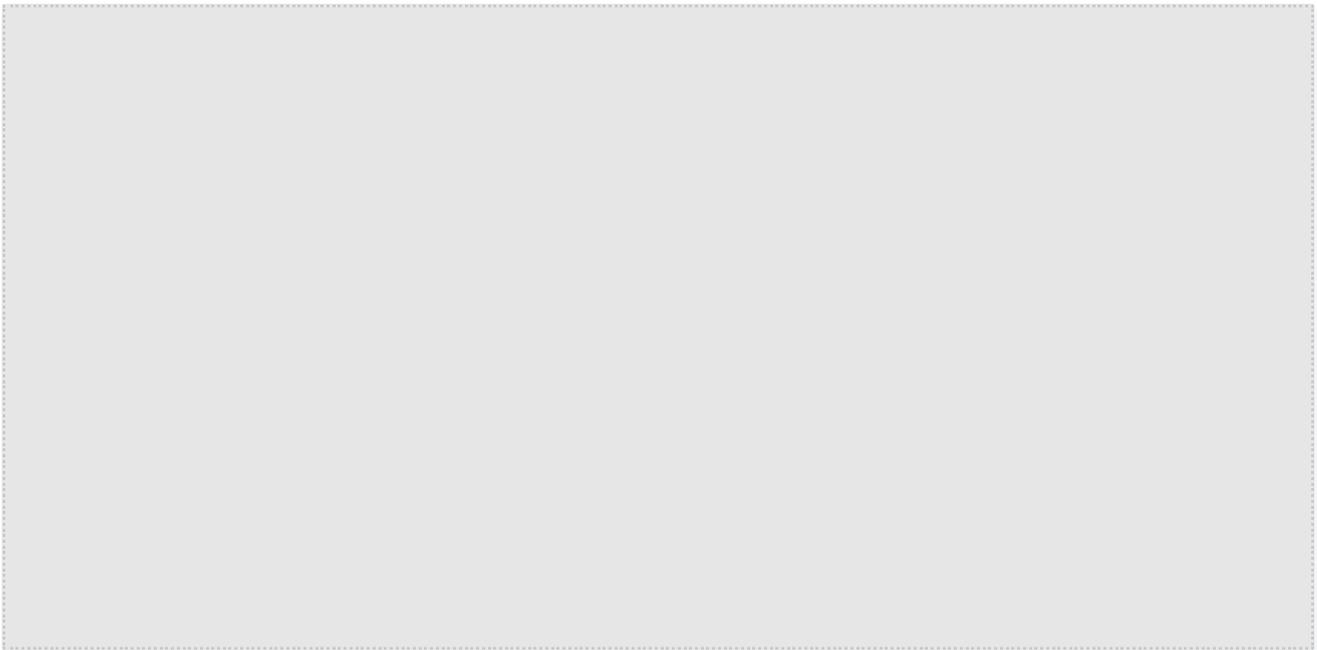
Bind Tencent Cloud TCHouse-P service

Enter the Tencent Cloud TCHouse-P control panel to view the cluster access address
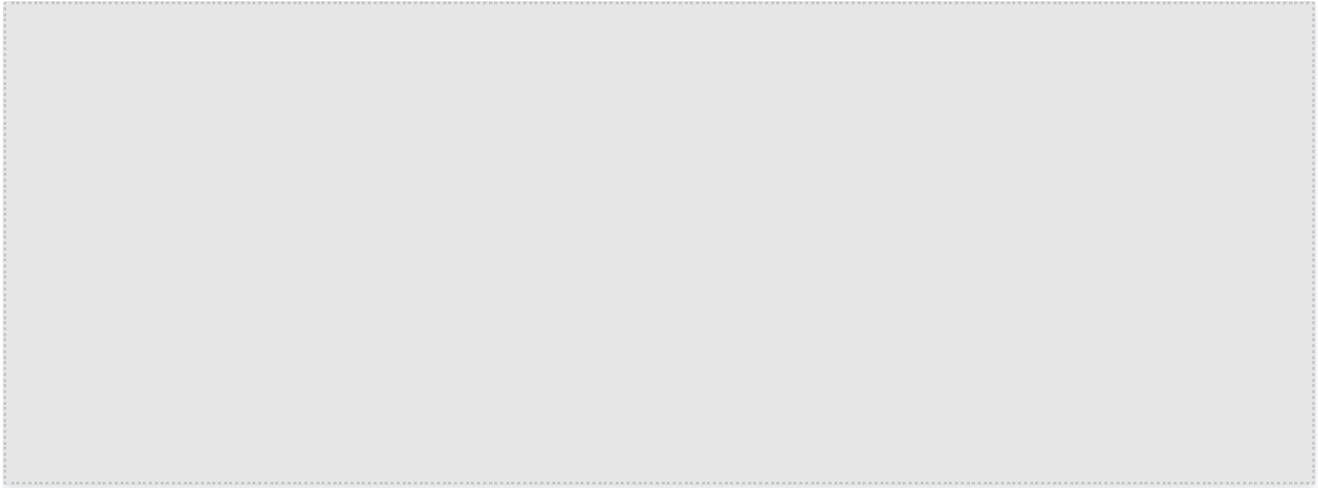
Enter the CLB control panel and click "Bind"

Bind Tencent Cloud TCHouse-P service

1. Select the target type as **"IP Type"**.

2. Enter **IP** and **Port**.

3. Click **"Confirm"**.
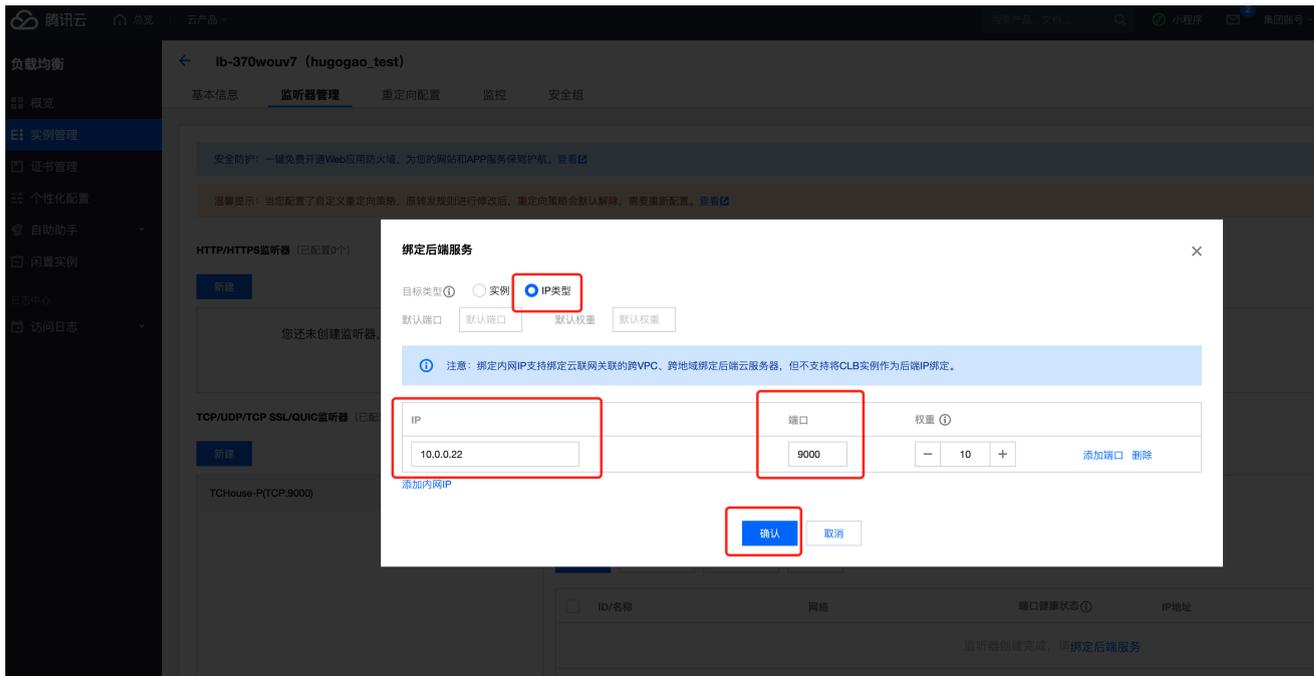
Access Tencent Cloud TCHouse-P via the CLB VIP

Managing Resource Queue

Last updated：2024-08-22 15:37:50

Operation scenarios

When using Tencent Cloud TCHouse-P, a single complex query may consume too many resources and affect other users' queries or computing. You can utilize a resource queue to limit system resource usage by a single user or query statement.

Operation step

1. Log in to **Tencent Cloud TCHouse-P**, in the Cluster list click **Manage** to enter the Cluster details page.

2. Select **Configuration** > **Resource Queue**, click **Create Resource Queue**, and configure parameter values. For details on the function of each parameter, refer to **Greenplum Resource Queue**.

3. Click **Confirm**. After the resource queue is created, you can specify it to limit resources during querying.

Terminating Clusters

Last updated：2024-08-22 15:38:15

Operation scenarios

When you no longer need a pay-as-you-go cluster, you can delete it through Tencent Cloud TCHouse-P . For refunds of monthly subscribed clusters, please refer to Refund Instructions .

> ⚠ **Note**
> Proceed with caution when deleting a cluster, as all the data will be deleted.

Operation step

1. In the cluster list, click **Manage** to enter the cluster details page
2. In the top-right corner, click **Terminate** to enter the termination page.

3. Click **Confirm**, and the pay-as-you-go cluster will enter the termination status.

> ⚠ **Note**
> Pay-as-you-go and monthly subscribed clusters are terminated on different pages.
> - A pay-as-you-go cluster is terminated after confirmation. Once confirmed, the termination will start, and data cannot be recovered.
> - You will be redirected to the order refund page to terminate a monthly subscribed cluster. After the refund is confirmed, the cluster will be isolated and then retained for seven days, during which you can renew it to use it again. Note that you can only renew it once. If the cluster is not renewed, data will be cleared after seven days.

Accessing Data Warehouse Connecting to Database

Last updated: 2024-08-22 15:38:50

By default, only CVM instances in the same VPC subnet can access Tencent Cloud TCHouse-P clusters. If you need to access the data warehouse cluster over the public network, apply for **Public network address**.
After creating a cluster, you need to use the database client to connect to the database before you can use the database services. Connect to the database with the client tool psql as instructed below.

1. Get the cluster access address: Connect to the database through the IP and port number in the JDBC URL of the cluster.
2. Connect to the cluster database: Install the client and connect to the cluster database.

Prerequisites

1. You have obtained the DMC admin password for the Tencent Cloud TCHouse-P cluster. The DMC admin password is the administrator account password set when the cluster was created.
2. You have obtained the IP, port number, VPC, and subnet information to access the created Tencent Cloud TCHouse-P cluster.

Getting Cluster Access Address and Local Network Information

Select the corresponding cluster. Detailed information is shown in the figure: VPC Network: vpc-aejsd98p, Subnet: subnet-83knqldq. The connection IP to Snova is 10.0.6.10, and the port is 5432. The login account is lambuser.

Connecting to Cluster Database on Command Line

In the obtained VPC Network: vpc-aejsd98p, Subnet: subnet-83knqldq, select a CVM server (if none exists, purchase one). Log in to the server and run the following commands to install the PostgreSQL client.

```
yum install -y postgresql.x86_64
```

Run the following SQL command and enter the password set during cluster creation to log in.

```
psql -h10.0.6.10 -p5432 -dpostgres -Ulambuser
```

Connecting to Database with JDBC

Get the JDBC officially provided by PostgreSQL, **Download address**.
Or add the following configuration in the pom.xml file:

```
<dependencies>
        <dependency>
            <groupId>org.postgresql</groupId>
```

```
            <artifactId>postgresql</artifactId>
            <version>42.2.2</version>
        </dependency>
</dependencies>
```

Sample code

```java
package com.qcloud.snova_conn;

import java.io.InputStream;
import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.PreparedStatement;
import java.sql.ResultSet;
import java.sql.ResultSetMetaData;
import java.sql.Statement;
import java.sql.Timestamp;
import java.util.ArrayList;
import java.util.List;
import java.util.Properties;
import java.sql.Connection;
import java.sql.ResultSet;
import java.sql.SQLException;
import com.yammer.metrics.core.Meter;

public class SnovaConn {
    /*
     * args: vip vport user pwd
     */
    public static void main(String[] args) throws ClassNotFoundException, SQLException {

        if (args.length < 4){
                System.out.println("args err");
                return;
        }

        String vip =  args[0];
        String vport =  args[1];
        String userName =  args[2];
        String userPwd =  args[3];

        System.out.printf("vip:%s, vport:%s, userName:%s, userPwd:%s\n",vip, vport, userName, userPwd);
        String jdbcUrl = "jdbc:postgresql://" + vip+":"+vport+"/maxluo";
        System.out.printf("jdbcUrl:%s \n",jdbcUrl);

        Class.forName("org.postgresql.Driver");
        Connection snova = DriverManager.getConnection(jdbcUrl,userName,userPwd);
        Statement st = snova.createStatement();
        ResultSet rs = st.executeQuery("select * from test;");
        while (rs.next()) {
                System.out.print(rs.getString(1));
                System.out.print("\n");
        }
        rs.close();
        st.close();
    }
}
```

pom.xml configuration

```xml
<project xmlns="http://maven.apache.org/POM/4.0.0" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://maven.apache.org/POM/4.0.0 http://maven.apache.org/xsd/maven-4.0.0.xsd">
<modelVersion>4.0.0</modelVersion>

<groupId>com.qcloud</groupId>
<artifactId>snova-conn</artifactId>
<version>0.0.1-SNAPSHOT</version>
<packaging>jar</packaging>

<name>snova-conn</name>
<url>http://maven.apache.org</url>

<properties>
    <project.build.sourceEncoding>UTF-8</project.build.sourceEncoding>
    <project.reporting.outputEncoding>UTF-8</project.reporting.outputEncoding>
    <maven.compiler.source>1.8</maven.compiler.source>
    <maven.compiler.target>1.8</maven.compiler.target>
</properties>

<dependencies>
    <dependency>
        <groupId>mysql</groupId>
        <artifactId>mysql-connector-java</artifactId>
        <version>5.1.40</version>
    </dependency>
    <dependency>
        <groupId>org.postgresql</groupId>
        <artifactId>postgresql</artifactId>
        <version>42.2.2</version>
    </dependency>
    <dependency>
        <groupId>com.microsoft.sqlserver</groupId>
        <artifactId>mssql-jdbc</artifactId>
        <version>6.4.0.jre8</version>
    </dependency>
    <dependency>
        <groupId>com.yammer.metrics</groupId>
        <artifactId>metrics-core</artifactId>
        <version>2.2.0</version>
    </dependency>
    <dependency>
        <groupId>ch.qos.logback</groupId>
        <artifactId>logback-classic</artifactId>
        <version>1.1.9</version>
    </dependency>
</dependencies>

<build>
    <plugins>
        <plugin>
            <artifactId>maven-assembly-plugin</artifactId>
                <configuration>
                <descriptorRefs>
                <descriptorRef>jar-with-dependencies</descriptorRef>
                </descriptorRefs>
                </configuration>
            <executions>
                <execution>
                <id>make-assembly</id>
                <phase>package</phase>
```

```xml
                    <goals>
                    <goal>single</goal>
                    </goals>
                    </execution>
                </executions>
            </plugin>
            <plugin>
                <groupId>org.apache.maven.plugins</groupId>
                <artifactId>maven-jar-plugin</artifactId>
                <configuration>
                    <excludes>
                        <exclude>*.properties</exclude>
                        <exclude>*.xml</exclude>
                        <exclude>*.json</exclude>
                        <exclude>*.sh</exclude>
                    </excludes>
                </configuration>
            </plugin>
            <plugin>
                <groupId>org.apache.maven.plugins</groupId>
                <artifactId>maven-dependency-plugin</artifactId>
                <executions>
                    <execution>
                        <id>copy-dependencies</id>
                        <phase>package</phase>
                        <goals>
                            <goal>copy-dependencies</goal>
                        </goals>
                        <configuration>
                            <type>jar</type>
                            <includeTypes>jar</includeTypes>
                            <outputDirectory>
                                ${project.build.directory}/lib
                            </outputDirectory>
                        </configuration>
                    </execution>
                </executions>
            </plugin>
        </plugins>
    </build>
</project>
```

Then use Maven to package and generate the jar file, and upload it to any CVM server in the VPC subnet where the Tencent Cloud TCHouse-P cluster is located.
Run the following command to install JDK.

```
yum install java
```

Run the following command:

```
java -cp  snova-conn-0.0.1-SNAPSHOT-jar-with-dependencies.jar  com.qcloud.snova_conn.SnovaConn  10.0.8.5
5436  lambuser  lambpwd11
```

> ⚠ **Note**
> The VIP and port number are the address to access the Tencent Cloud TCHouse-P cluster, and the username and password are those entered during cluster creation as detailed above.

Create a database and data table on the command line and insert a certain amount of data.

The query results are as follows, data can be read from the pre-established table 'test' in the database 'maxluo':
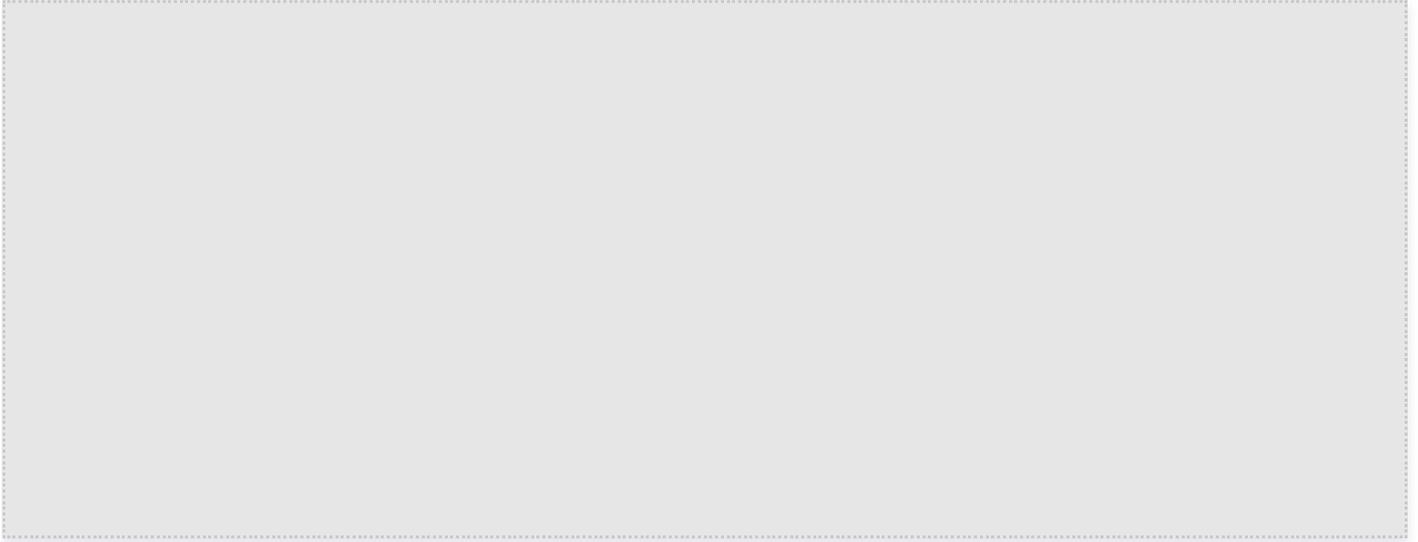
Managing User Permission

Last updated: 2024-08-22 15:39:39

Role Overview

In Tencent Cloud TCHouse-P, roles are used to manage access to databases. The concept of roles actually includes both users and groups. A role can be a regular database user (user) or a database user group (group). Roles can own database objects (such as tables and views) and assign access to these objects to other roles.

When creating a cluster, users are prompted to set an initial username and password. This initial user is an "administrator", who has the permissions to create users, databases, and log in. After the cluster is created, you can connect to the database using the "administrator" account. Generally, the administrator has the highest level of permissions, which means this account should be used by as few people as possible. Therefore, you can create other users with the administrator account and grant them the necessary permissions. For specific authorization, refer to User Groups and Object Permission Management. You can also create databases and other objects, refer to Definition of Databases. For how to log in to the database, refer to Connecting to the Database.

Creating user

Roles are divided into user and group. Typically, a role at the user level (hereafter referred to as user) has permissions to log in to the Tencent Cloud TCHouse-P database and initialize a session. Therefore, when creating a user, you must grant it the LOGIN permission. For example:

```
CREATE role jsmith with LOGIN;
```

Through the above operation, a user with LOGIN permission is created. This user can connect to the database. Tencent Cloud TCHouse-P provides CAM users with other permissions in addition to LOGIN. You can grant these permissions when creating a role using the CREATE ROLE statement.

Permissions	Effect	Default Value
SUPERUSER Ι NOSUPERUSER	Superuser permission. Only superusers can create other superusers	NOSUPERUSER
CREATEDB Ι NOCREATEDB	Creates databases	NOCREATEDB
CREATEROLE Ι NOCREATEROLE	Creates and manages roles	NOCREATEROLE
INHERIT Ι NOINHERIT	Determines the permissions a user inherits from the group to which the user belongs	INHERIT
LOGIN Ι NOLOGIN	Connects to databases, which is granted to users but not groups	NOLOGIN
CONNECTION LIMIT	Limits the number of concurrent connections to a database. -1 means no limit	-1
CREATEEXTTABLE Ι NOCREATEEXTTABLE	Privilege to create external tables	NOCREATEEXTTABLE
PASSWORD	Sets the password during user creation	No
VALID UNTIL 'timestamp'	Password expiration time	No
RESOURCE QUEUE 'name'	The name of the resource queue to which the created query is scheduled after a user establishes a connection	pg_default

In addition to granting permissions when creating users, you can also reassess permissions using the ALTER ROLE syntax after creation. For example:

```
ALTER role jsmith with CREATEROLE;
```

User Group

A group, or user group, is a special role that is not granted the LOGIN permission. A group is usually set up as a combination of frequently used permissions, which can be granted or revoked as a whole to or from a user.
You can create a group that is granted a combination of permissions by using the following statement.

```
Create role, Create DB, Cannot login;
```

You can easily establish or revoke other users' subordinate relationships with this user group using the GRANT TO or REVOKE FROM statement. Users with subordinate relationships inherit permissions from the user group.

- Example of the GRANT TO statement:

User jsmith is assigned to the user group manager.

- Example of the REVOKE FROM statement:

The association of user jsmith with the user group manager is removed.

Object Permission Management

When an object (database, table, schema, function, etc.) is created, it must belong to an owner, which is usually the user who runs the object creation statement. Initially, only the owner has all the permissions to manipulate the object; for example:

```
GRANT INSERT ON test TO  jsmith;
```

We can use the above statements to grant the INSERT permission on test to the user jsmith, and similarly, revoke it using the REVOKE FROM statement.
Similarly, you can reassign all objects owned by a user to another user using the REASSIGN OWNED statement. For example:

```
SET ROLE jsmith;    // Switch to jsmith user
CREATE TABLE jsmithtest (age int, id int); // Create a new table
SET ROLE gpadmincloud;       // Switch back to the superuser
REASSIGN OWNED BY jsmith TO lambuser;   // Transfer all objects owned by jsmith to lambuser.
```

The objects owned by a superuser cannot be transferred to other users, because some of the objects also belong to the system. Therefore, you need to create a table as a non-superuser.
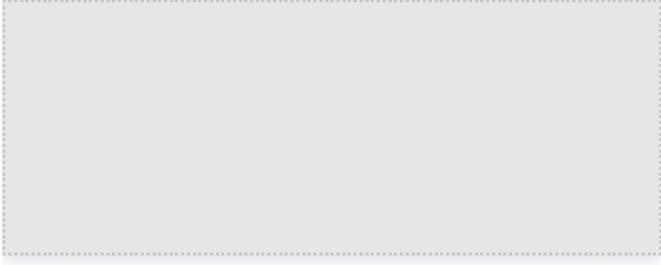
Completed the transfer of ownership from jsmith to lambuser.

Defining Database

Last updated: 2024-08-22 15:40:03

Database Creation and Management

In Tencent Cloud TCHouse-P, you can create your own database objects.

1. If you want to create a database for your use, first refer to Managing User Permission to create and authorize users. Log in to, then use the CREATE DATABASE statement to create the database. However, before creating the database, ensure that the user logged in has the CREATEROLE permission. For permission issues, see Managing User Permission . Example of database creation is as follows:

```
CREATE DATABASE test;
```

Through `\l` , you can list all the databases.

2. During the database creation process, you often choose a database template to create a new database. The default database template is empty. If there are any objects in the template database, the newly created database will also have corresponding objects. You can also specify a template to create. For example, first create a table in test using the following statement.

```
create table ttable (age int, id int);
```

Then create the database test2 using test as a template.

```
CREATE DATABASE test2 TEMPLATE test;
```

Switch to test2 and you will see that test2 also has ttable. Therefore, avoid creating any objects in template1, otherwise, the database created using template1 as a template will have corresponding objects. You can use `\d` to view all tables in the selected database.

```
test=# create database test2 template test;
CREATE DATABASE
test=# \c test2
You are now connected to database "test2" as user "gpadmincloud".
test2=# \d
            List of relations
 Schema |  Name  | Type  |    Owner     | Storage
--------+--------+-------+--------------+--------
 public | ttable | table | gpadmincloud | heap
(1 row)
```

3. You can list all databases using `\l` .
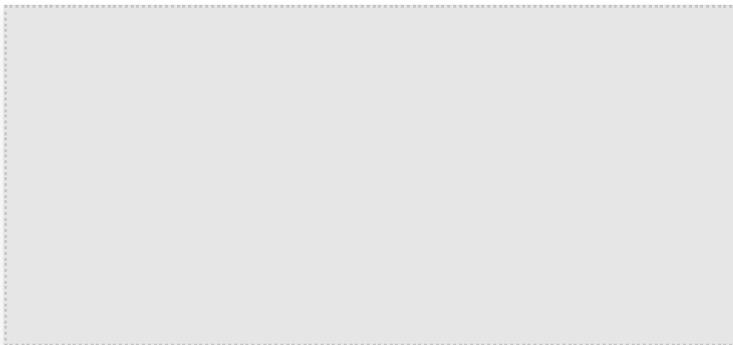
4. You can delete a database using the DROP DATABASE command. To perform the deletion, ensure that the user logged in is a superuser or a normal user with database deletion rights. Additionally, the database can only be deleted if the connection count is 0. For example:

```
test2=# DROP DATABASE test2;
ERROR:  cannot drop the currently open database
test2=# \c test
You are now connected to database "test" as user "gpadmincloud".
test=# DROP DATABASE test2;
DROP DATABASE
```

You can see that when the user selects test2, the connection count for test2 is necessarily greater than or equal to 1. Switch to test, and then you can normally delete test2.

Schema Creation and Management

In Tencent Cloud TCHouse-P, a schema as a logical concept exists to provide a more detailed division of database space. Each database has a schema named public when created. Inside a database, creating tables with the same name is not allowed. However, creating tables with the same name in different schemas is permissible. The database system identifies a table as database.schema.table. In addition, different databases can have schemas with the same name.

1. Creating Schema.

```
CREATE SCHEMA testschema;
```

2. Specify the schema to create objects.
   When creating tables, functions, and other objects, you can add the schema prefix to indicate creating objects in different schemas. Otherwise, it defaults to the public schema. For example:

```
CREATE TABLE testschema.test;
```

3. Set the priority of schemas.

```
ALTER DATABASE test set search_path to testshcema,public;
```

   Set the testschema of database test as the public schema, indicating that the highest priority schema is testschema. Without any schema prefix, it first matches testschema.

4. Schema Switch.

```
SET search_path TO public;
```

   If currently in the testschema schema, this statement can switch the schema to public.

5. Deletion Mode.

```
DROP SCHEMA testschema;
```

Table Creation and Management

1. Set Table and Column Constraints.

- CHECK constraint: This constraint can specify that a column's data must satisfy a certain expression. For example:
  ```
  CREATE TABLE products (product_no int, name text, price int CHECK(price > 0));
  ```

- NOT NULL constraint: This constraint can specify that a column's data cannot be null. For example:

```
CREATE TABLE products (product_no int NOT NULL, name text NOT NULL, price int CHECK(price > 0));
```

2. Data Distribution Policies.
   For a distributed database warehouse, the best processing performance can be achieved when the amount of data stored in each node is the same. If data distribution is unbalanced, nodes with more data will take longer to complete the query process, leading to decreased performance for the entire query.

- Hash Distribution: Using the DISTRIBUTED BY syntax, you can specify hash distribution when creating a table. The hash distribution method will combine all keys specified for hash distribution and determine the data distribution result using a hash algorithm. The statement is as follows:

```
CREATE TABLE test (id int, age int) DISTRIBUTED BY (id);
```

- Random Distribution: Using the DISTRIBUTED RANDOMLY syntax, you can specify random distribution when creating a table. As the name suggests, the data distribution result is determined randomly. The statement is as follows:

```
CREATE TABLE test (id int, age int) DISTRIBUTED RANDOMLY;
```

Columns with PRIMARY KEY or UNIQUE must specify one of the data distribution policies. For columns without PRIMARY KEY and UNIQUE attributes, the first column is used as a reference for data distribution by default. The default data distribution policy is hash distribution.

View Creation and Management

A view is a logical concept; unlike a table, a view does not have an actual data structure on the hard disk.

1. Create View.

```
CREATE VIEW testview AS SELECT * FROM ttable where age=28;
```

Create a view named testview that includes all rows from ttable where "age=28".

2. Drop View.

```
DROP VIEW testview;
```

Administration Data

Last updated: 2024-08-22 15:40:29

Inserting Data

1. Insert data corresponding to the column name.

```
INSERT INTO products (name, price, product_no) VALUES ('cheese', 99, 1);
```

2. Insert data in the order of the column names defined in the table.

```
INSERT INTO products VALUES(2, 'chesse', 99);
```

3. Insert multiple data records at once.

```
INSERT INTO products VALUES (3, 'a', 1), (4, 'b', 2), (5, 'c', 3);
```

4. Import data through an external table, as instructed in Using External Table to view the method for importing via external tables.
5. Import data from TencentDB via plugins as instructed in Importing External Data.
6. Insert through the COPY command. First, you need to log in to the **Database system**, select the database, create the corresponding table, and then use the copy command to insert data from the specified file filename with the specified delimiter ',' into tablename. The command is as follows:

```
COPY tablename  FROM 'filename'  WITH DELIMITER ',';
```

Updating Data

Update the data of the columns in rows that meet the WHERE condition to the specified value. An example is as follows:

```
UPDATE products SET price = 10 where product_no = 3;
```

Deleting Data

Delete the rows that meet the WHERE condition. An example is as follows:

```
DELETE FROM products where price = 3;
```

Delete all data in the table as shown below:

```
DELETE FROM products;
```

Querying Data

After accessing the database via the Accessing Data Warehouse method, you can perform data query operations. An example is as follows:

1. First, enter the specified DB, such as the test database.

```
\c test;
```

2. Create a table test.

```
create table test(a1 int);
```

3. Insert the data.

```
insert into test values(3),(4);
```

4. Query the data.

```
select * from test;
```

Monitoring and Alarming
Alarm Configuration

Last updated：2024−08−22 15:41:15

Background

Tencent Cloud TCHouse−P provides a Performance Monitoring Dashboard, allowing users to observe the historical and current status of the performance metrics of each node in the cluster. Tencent Cloud TCHouse−P offers an alarm notification feature to inform users of sensitive metrics exceeding thresholds, such as the disk usage of each node.

Performance Monitoring

Enter Tencent Cloud TCHouse−P , click on **Cluster Name** in the cluster list to go to the cluster details page. In **Performance Monitoring**, you can view various metrics of the cluster. When there are multiple nodes, you can select the cluster node you want to view in **Node Dimension**.
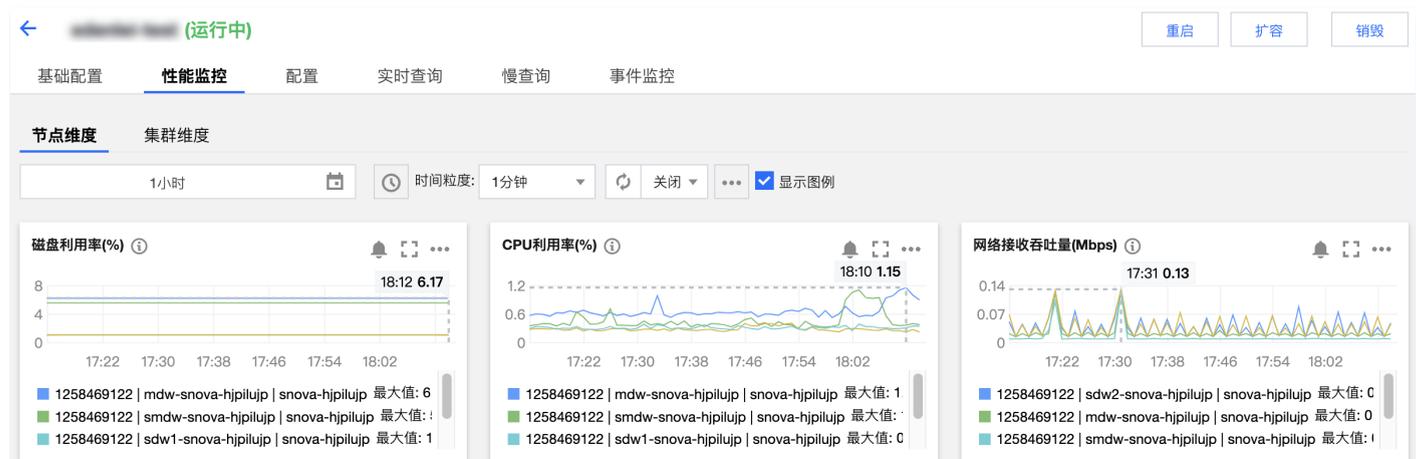
Currently, Tencent Cloud TCHouse−P provides metrics such as the number of connections, CPU utilization, memory usage, network receive throughput, network send throughput, write IOPS, read IOPS, disk space utilization, read throughput, write throughput, read latency, and write latency.

Alarm Access

Tencent Cloud TCHouse−P alarms are categorized into three types: Cluster Monitoring, Primary Node Monitoring, and Compute Node Monitoring. These three alarm types notify users across three dimensions.

Creating alarm policy

Enter TCOP Console , in **Alarm Management > Alarm Configuration > Alarm Policy**, click **Create** to add a new alarm policy. When creating an alarm policy, **Policy Type** can be selected as cdwpg data warehouse − cluster monitoring, cdwpg data warehouse − primary node monitoring, and cdwpg data warehouse − compute node monitoring. This section takes compute node monitoring as an example.

1. Select **Policy Type** as cdwpg data warehouse – compute node monitoring.
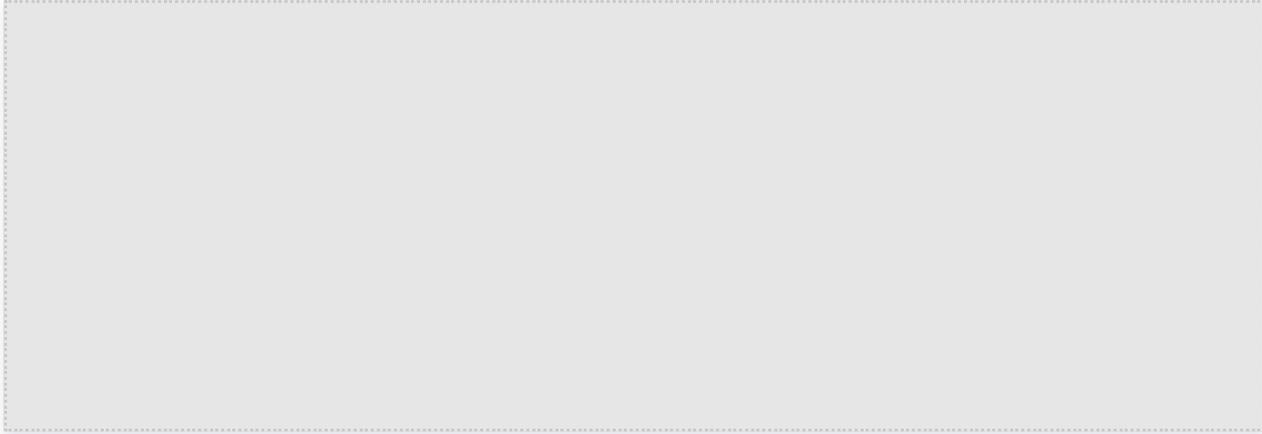
2. Set the **Alarm Target**, you can choose different compute nodes groups through the dropdown menu.

3. Configure the **Trigger Condition**, through **selecting a template** or **manual configuration**.

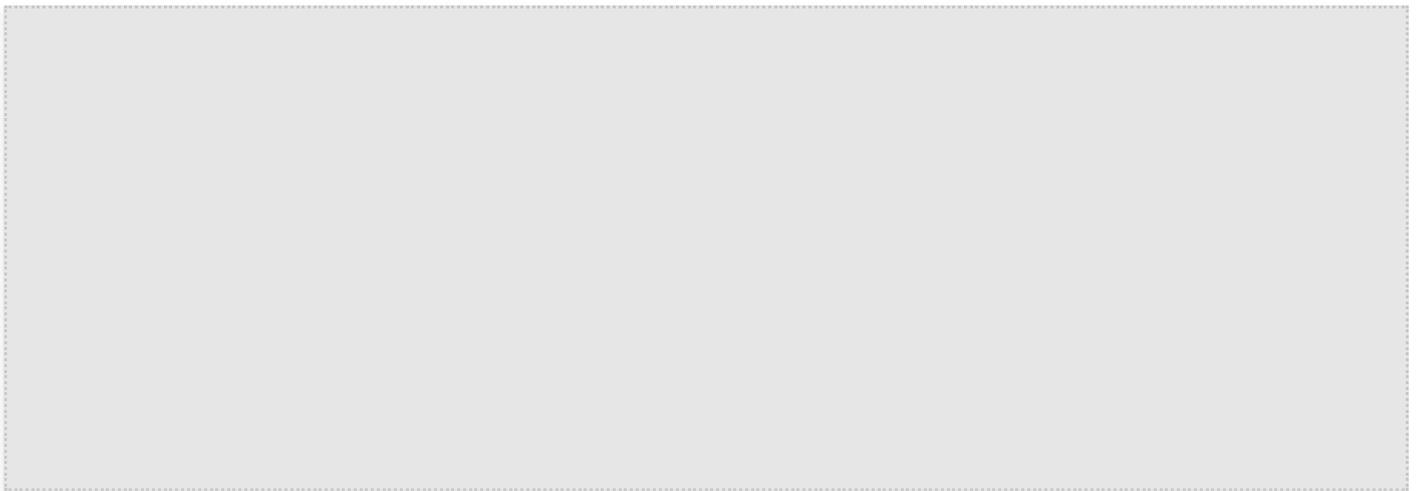- If selecting trigger condition: **Select Template**, you can click on **Add Configuration Trigger Condition** to configure the threshold for triggering an alarm for each metric. Once the metric reaches the threshold condition, the system will send you an alarm message. You can also click on **Modify Template** to modify an existing template.

- If selecting **Manual Configuration**, you can add other metrics that you are interested in, by setting thresholds to establish different alarm thresholds and notification cycle policies for different metrics.

4. Configure the **Notification Template**, you can click on **Select Template** to choose an existing template or click on **Create Template** to create a new notification template.
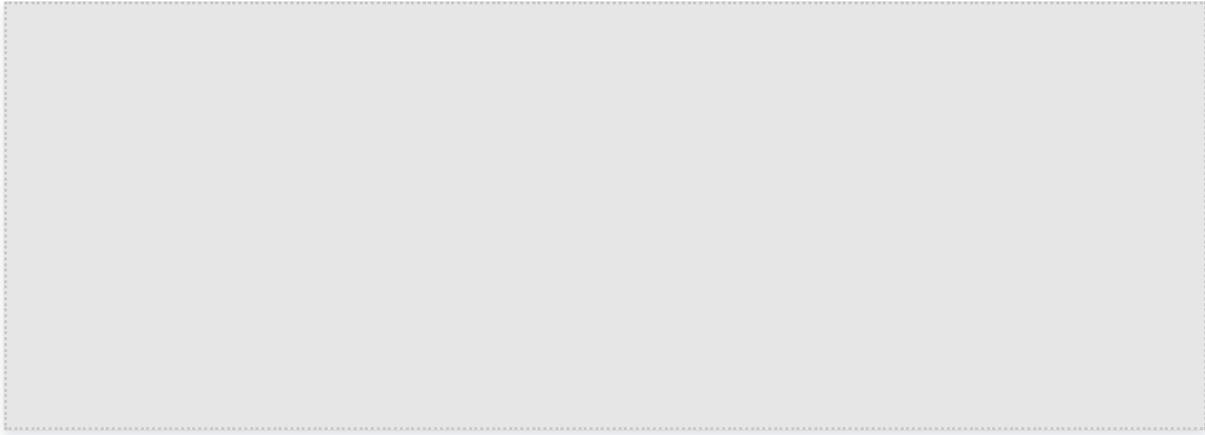
When creating a new notification template, configure the **Receiving Object** for development and operations personnel who are interested in or need to handle the alarm information. You can choose to alert through Email, SMS, and WeChat.
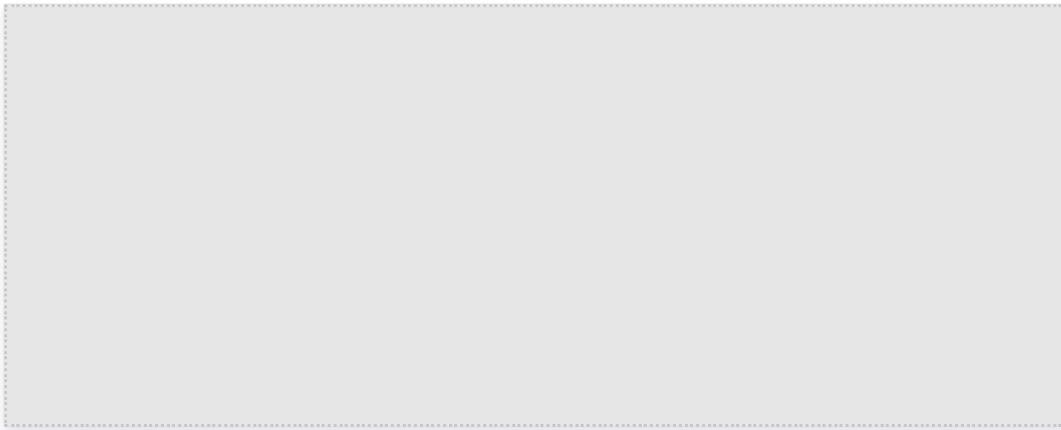
Account Management

Last updated: 2024-08-22 15:41:51

Tencent Cloud TCHouse-P provides a visual interface in the console, enabling convenient and efficient operations such as account management, password management, and authorization for clusters.

Account Management

1. Log in to Tencent Cloud THouse-P console , click on the target cluster **ID/Name**, and find **Account Management** in the left-hand menu after navigating.
2. On the Account Management page, features for resetting the password and authorization are provided.
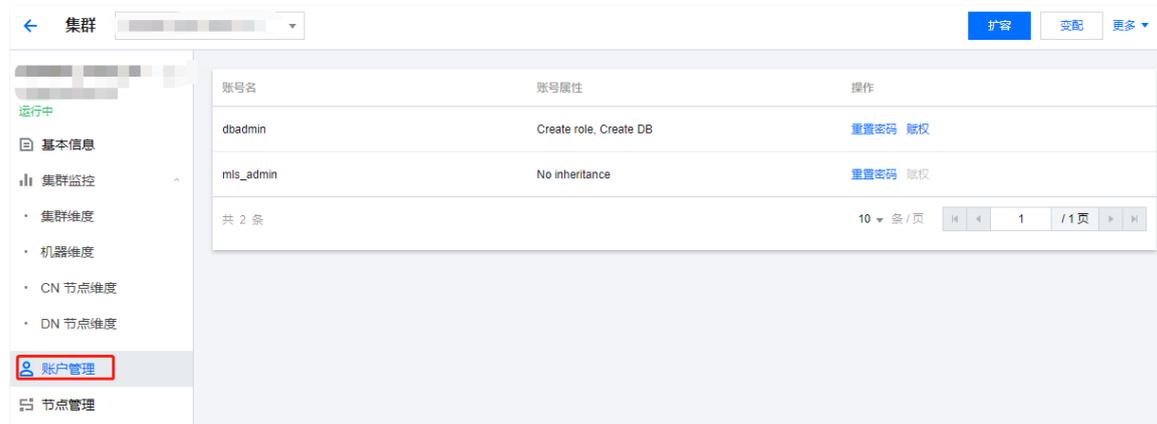
Resetting the password

1. Tencent Cloud TCHouse-P does not support viewing existing account passwords; you can only reset the password through the console.
2. If you forget the password, it is recommended to perform the following operations:
   - If you forget the Admin account password, please contact us through a ticket to reset the password.
   - If you forget a sub-account password, you can reset it through the console.
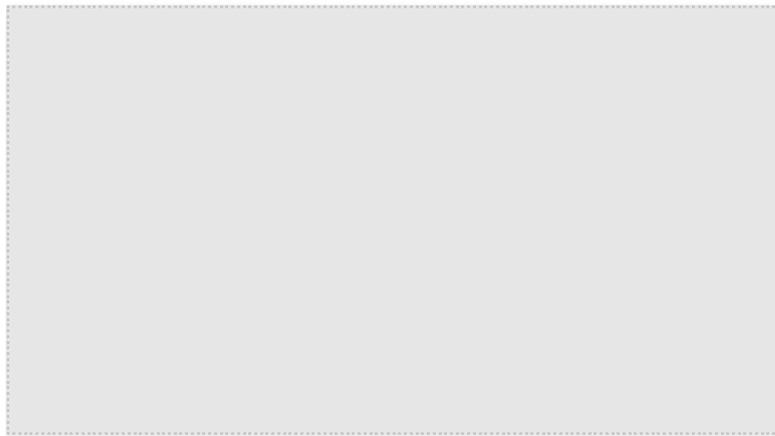
> ⓘ **Note:**
> Password setting rules: Start with letters or numbers, include at least three of the following: lowercase letters, uppercase letters, numbers, and special characters (!@#%^*), do not start with "/", and between 8 to 16 characters in length (recommended 12 or more).

Account Authorization

The account authorization operation is currently open only for dbadmin, allowing dbadmin to authorize the use of the hdfs_fdw feature.

账户赋权 ⓘ   ✕

数据库 *   请选择数据库   ▼

🔍

☐ postgres

☐ product_online_reports

数据库 *   请选择数据库   ▼

Cloud Access Management
Permission Management

Last updated：2024–08–22 15:43:34

Concepts

Cloud Access Management (CAM) is a web–based Tencent Cloud service that helps you securely manage the access permissions of resources under your Tencent Cloud account. With CAM, you can create, manage, and terminate users (groups) and use identity and policy management to control user access to Tencent Cloud resources.

Granting Access

Granting access permission means that the user can determine the combination of control capabilities for specific actions on specific resources under specific conditions. Therefore, to describe an access permission behavior, four elements are usually included: **identity, resources, actions, and conditions (optional)**.

Access Authorization Elements

Tencent Cloud identity

When a user registers for a Tencent Cloud account, the system creates a root account identity for logging in to Tencent Cloud services. The Tencent Cloud root account can use the user management feature to manage classified users with different roles. User types include **Collaborator, Message Recipient, Sub–user, and role**. For specific definitions, see Identity Management and Glossary .

Tencent Cloud TCHouse–P cluster resources

The resources of Tencent Cloud TCHouse–P refer to the TCHouse–P cluster. Access control is also targeted at the TCHouse–P cluster. Typically, in the console, we can see the identifier of Tencent Cloud TCHouse–P, as shown below:

▶ snova-28fg7yl3 ←	nc.large	广州四区	包年包月	运行中	0	2020-04-20 14:53:56	管理

where snova–28fg7yl3 is the unique identifier of the cluster, which can also be understood as the identifier of Tencent Cloud TCHouse–P resources.

Operations of Tencent Cloud TCHouse–P cluster

The operations on the cluster refer to the actions performed by the user in the console of Tencent Cloud TCHouse–P. Basically, each operation can be mapped to a TencentCloud API, such as deleting a cluster or viewing detailed information of a cluster, each having an action identifier. Access control can be performed for actions (read, write) during access control.

Principle of least privilege

When granting authorization, please specify the scope of the permission clearly, specifying what **designated user** can perform under **what conditions**, perform **what actions**, and access **what resources**.

Setting Policy

Last updated: 2024-08-22 15:43:57

Overview

An access policy that employs the JSON-based access policy language is used to grant access to Tencent Cloud TCHouse-P Cluster resources. You can authorize a specified principal to perform actions on a specified Tencent Cloud TCHouse-P Cluster resource through the access policy language.

The access policy syntax describes the basic elements and usage of the policy. For the description of the policy syntax, see CAM Policy Management.

Elements in Access Policy

An access policy contains the following elements with basic meanings:

- **statement**: Describes the details of one or more permissions. This element contains a permission or permission set of other elements such as effect, action, resource, and condition. One policy has only one statement element.
- **effect**: Describes the result of a statement as either "allow" or "explicit deny", including both allow and deny outcomes. This element is required.
- **action**: Describes the allowed or denied action. An action can be an API (prefixed with name) or a feature set (a specific set of APIs, prefixed with permid). This element is required.
- **resource**: Describes the details of authorization. A resource is described in a six-segment format. Detailed resource definitions vary by product. This element is required.
- **condition**: Describes the condition for the policy to take effect. A condition consists of operator, action key, and action value. A condition value may contain information such as time and IP address. Some services allow you to specify additional values in a condition. This element is optional.

Element Usage

Specifying effect

If access to a resource is not explicitly granted (allowed), it is implicitly denied. Explicit denial (deny) of access to a resource ensures that a user cannot access the resource, even if other policies grant access. The following is an example of specifying an allow effect:

```
"effect" : "allow"
```

Specifying action

Tencent Cloud TCHouse-P defines a set of actions in the policy for the console. The specified actions are classified by nature into read-only interfaces (cdwpg:Describe*) and all interfaces (cdwpg:*).

Below is an example of specifying the allowed operations:

```
"action": [
  "name/cdwpg:Describe*"
]
```

Specifying resource

The resource element describes one or more action objects, such as Tencent Cloud TCHouse-P Cluster resources, etc. All resources can be described using the following six-segment format.

```
qcs:project_id:service_type:region:account:resource
```

The parameters are described as follows:

Parameter	Description	Is it required
qcs	Tencent Cloud service abbreviation, which indicates a service of Tencent Cloud	Yes
project_id	Project information, which is only used to enable compatibility with legacy CAM logic and generally can be left empty	No
service_type	This is for cdwpg	Yes
region	Region information	Yes
account	Root account information of the resource owner, i.e., root account UIN in the format of `uin/${OwnerUin}`, such as uin/100000000001	Yes
resource	Describes specific resource details, prefixed with cdwpg-instance	Yes

Below is a six-segment information of a Tencent Cloud TCHouse-P cluster:

```
"resource":["qcs::cdwpg:ap-guangzhou:uin/100000000001:cdwpg-instance/snova-73jingds"]
```

Specifying condition

The access policy syntax allows you to specify conditions when granting permissions. In the scenario of Tencent Cloud TCHouse-P, it is mainly used to set tag authentication. The tag condition takes effect only for clusters bound with the tag. Below is a sample tag policy:

```
"condition": {
            "for_any_value:string_equal": {
                "qcs:tag": [
                    "jing&jingfdd"
                ]
            }
        }
```

This statement means that the policy includes resources with the Tag key as jing and value as jingfdd.

Practical Example

In the following example, the policy means that access is allowed to resources under UIN 1250000000 with the cluster ID snova-jidnshgdsh and resources bound with Tag key testkey and Tag value testvalue.

```
{
    "version": "2.0",
    "statement": [
        {
            "action": [
                "name/cdwpg:Describe*",
            ],
            "condition": {
                "for_any_value:string_equal": {
                    "qcs:tag": [
                        "testkey&testvalue"
                    ]
                }
            },
            "effect": "allow",
            "resource": [
                "qcs::cdwpg:ap-guangzhou:uin/1250000000:cdwpg-instance/snova-jidnshgdsh"
            ]
```

```
            }
        ]
    }
```

Granting Policy

Last updated：2024-08-22 15:44:35

Tencent Cloud TCHouse-P Preset Policy Management

To facilitate user authorization for sub-account resources, Tencent Cloud TCHouse-P has set up two preset policies in advance. Enter CAM Console, and search for "CDWPG" in the search box at the top right to see the following two policies:

Policies	Description
QcloudCDWPGFullAccess	Grants full administrative permissions for Tencent Cloud TCHouse-P
QcloudCDWPGReadOnlyAccess	Grants read-only administrative permissions for Tencent Cloud TCHouse-P

- To grant a user permissions to create and manage Tencent Cloud TCHouse-P instances, use the policy named QcloudCDWPGFullAccess for this user.
- To grant a user only query permissions for Tencent Cloud TCHouse-P clusters and related resources (VPC, security group, Monitor), but not permissions for operations such as create, delete, and modify, use the policy named QcloudCDWPGReadOnlyAccess for this user.

Tencent Cloud TCHouse-P Custom Policy

If preset policies cannot meet your needs, click Create Custom Policy to create a custom policy.
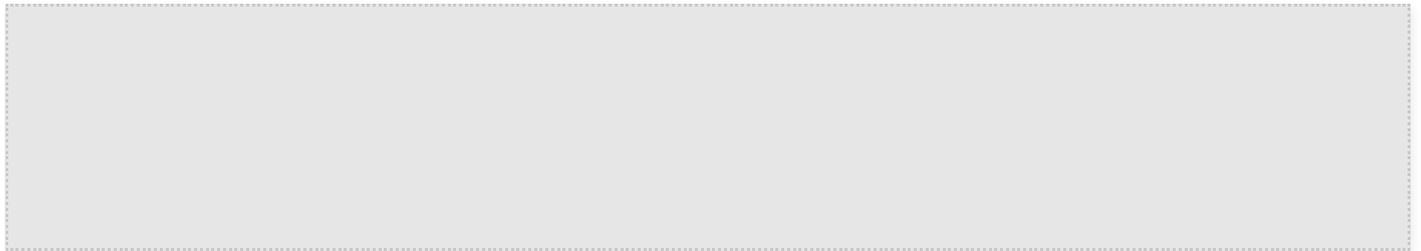
For the method of custom policy creation, see Policy Settings .

Policy Authorization

A configured policy can grant permissions by associating user groups or sub-users.

Resource Types Authorizable by Custom Policy

Resource-level permissions refer to the ability to specify which resources a user is allowed to perform actions on. Tencent Cloud TCHouse-P partially supports resource-level permissions, meaning that for Tencent Cloud TCHouse-P operations that support

resource-level permissions, you can control when a user is allowed to perform actions or use specific resources. The resource types that can be authorized in CAM are as follows:

Resource Type	Resource Description Method in Access Policies
Tencent Cloud TCHouse-P	`qcs::cdwpg:$region:$account:cdwpg-instance/*` `qcs::cdwpg:$region:$account:cdwpg-instance/$clusterId`

The table below introduces the API operations of Tencent Cloud TCHouse-P currently supporting resource-level permissions. When setting policies, enter the API operation name into action to control individual APIs. You can also use * as a wildcard when setting action.

API List Supporting Resource-level Authorization

Operations through API	Resource Path
ModifyClusterSize	Modifies the number of cluster nodes
DescribeClusters	Gets cluster details
DescribeRealtimeQuery	Gets real-time query details of a cluster
DescribeHistoryQuery	Gets historical query details of a cluster
AbortQuery	Aborts a cluster query
DescribeRealtimeQueries	Gets the list of real-time queries in a cluster
DescribeGpStatus	Gets the cluster database status
RebootCluster	Restarting Clusters
DescribeClusterStatus	Gets the cluster status
ModifyClusterSubnet	Modifies the cluster subnet
DescribeHistoryQueries	Gets the list of historical queries in a cluster
DeleteCluster	Deleting cluster
ModifyClusterUserPassword	Resets the cluster password
ModifyClusterBasic	Modifying cluster name
DescribeClustersStatistics	Gets the number of clusters
DescribeVpcLinks	Gets the VPC access link of a cluster
CreateVpcLink	Create VPC Access Link
DeleteVpcLink	Deletes a VPC access link
ExpandClusterSize	Scale Cluster
DescribeHbaConfigList	Gets the access address allowlist of a cluster
SetHbaConfigList	Modifies the access address allowlist of a cluster
DescribeClusterResourceQueueList	Queries the resource queue list of a database cluster
DescribeClustersLimit	Queries the resource limit configuration of a database cluster
HandlerResourceQueue	Manipulates a database resource queue
AdminClusterOutnetAddres	Manage public network access addresses
DescribeClustersNodesInfo	Gets the node information of a cluster

List of APIs not supporting resource-level authorization

For Tencent Cloud TCHouse API operations that do not support resource-level authorization, you can still grant users permission to use the operation, but the resource (resource) element in the policy statement must be set to *.

Operations through API	API Description
DescribeNodeConfigInfo	Gets the node model specification information
DescribeEvents	Gets the information of all cluster events
CreateCluster	Creating cluster
DescribeDbStatus	Gets the database status
DescribeZones	Get optional purchase regions
DescribeSegNodeMaxCount	Queries the maximum number of compute nodes
DescribeClusterExtend	Gets all Ops information of a cluster
DescribeResidual	Gets the resource status in a region
DescribeSpecResidual	Checks whether specific specifications are sold out
DescribeZonesResource	Get availability zone resource information
DescribeValidRegionAndZones	Gets valid regions and AZs