TaurusDB

Best Practices

Issue 01

Date 2025-07-11

HUAWEI CLOUD COMPUTING TECHNOLOGIES CO., LTD.

Trademarks and Permissions

Notice

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

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

Huawei Cloud Computing Technologies Co., Ltd.

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

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

Contents

1 From ECS-hosted MySQL to TaurusDB.. 1
1.1 Overview.. 1
1.2 Resource Planning.. 3
1.3 Operation Process... 4
1.4 Procedure.. 5
1.4.1 ECS-hosted MySQL Server... 5
1.4.1.1 Creating a VPC and Security Group... 5
1.4.1.2 Creating an ECS (MySQL Server).. 7
1.4.1.3 Installing a MySQL Database (Community Edition)... 8
1.4.1.4 Creating an ECS and Installing a MySQL Client on It... 10
1.4.2 Cloud Migration.. 10
1.4.2.1 Creating a TaurusDB Instance... 10
1.4.2.2 Creating a DRS Migration Task... 11
1.4.2.3 Checking the Migration Results.. 12
2 From Other Cloud MySQL to TaurusDB.. 14
2.1 Overview... 14
2.2 Resource Planning... 15
2.3 Creating a VPC and Security Group... 16
2.4 Creating a TaurusDB Instance.. 17
2.5 Configuring a MySQL Instance on Other Clouds... 18
2.6 Creating a DRS Migration Task.. 18
2.7 Checking Migration Results... 20
3 Enabling Read/Write Splitting... 22
3.1 User Authentication... 22
3.2 Connection Pool Configuration.. 25
3.3 Routing Read Requests to the Primary Node... 25

1 From ECS-hosted MySQL to TaurusDB

1.1 Overview

This practice describes how to install a MySQL database (community edition) on a Huawei Cloud ECS and create a TaurusDB instance, and use DRS to migrate data from MySQL to TaurusDB. With DRS, you can perform real-time migration tasks with minimal downtime.

Scenarios

- With the rapid increase of enterprise workloads, traditional databases have poor scalability and require distributed reconstruction.
- Building traditional databases requires purchasing and installing servers, systems, databases, and other software. Its O&M is expensive and difficult.
- Traditional databases are poor in complex queries.
- It is hard for traditional databases to smoothly migrate data with no downtime.

Prerequisites

- You have completed real-name authentication.
- Your account balance is at least $0 USD.

Solution Architecture

In this practice, the source database is an ECS-hosted MySQL instance and the destination database is a TaurusDB instance. **Figure 1-1** shows the deployment architecture when the ECS-hosted MySQL and TaurusDB instances are in the same VPC.

If the ECS-hosted MySQL and TaurusDB instances are not in the same VPC, you need to configure a **VPC peering connection** between the two VPCs. For details about the deployment architecture, see **Figure 1-2**.

Figure 1-1 Deployment architecture in the same VPC

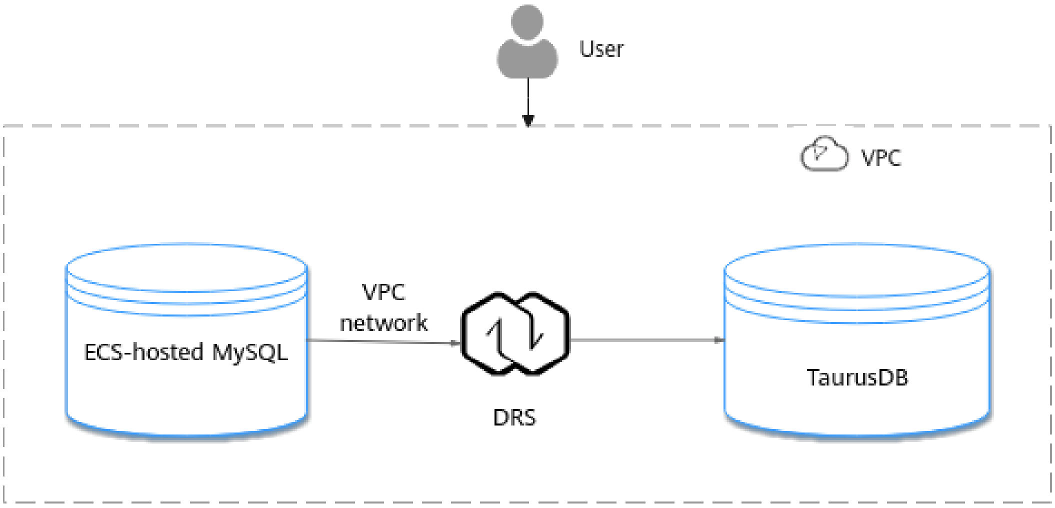
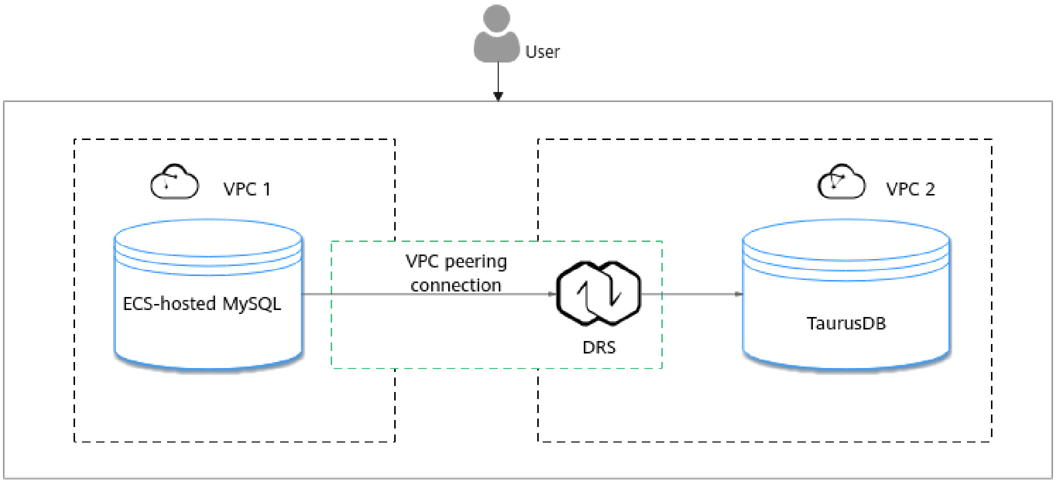


Figure 1-2 Deployment architecture in different VPCs



TaurusDB Advantages

- Robust performance: TaurusDB decouples compute from storage and uses a "log as database" architecture and remote direct memory access (RDMA). It can deliver seven times the performance of open-source MySQL for certain service loads.
- Elastic scaling: In addition to a primary node, you can add up to 15 read replicas for a DB instance within minutes. You can also scale up or down CPU and memory specifications for a DB instance as needed.
- High reliability: DB instances can be deployed across AZs and there are three data copies under the shared distributed storage layer. A DB instance failover can be complete within seconds with a zero RPO.
- High security: With shared distributed storage, TaurusDB ensures zero data loss and fault recovery within seconds. VPCs, security groups, SSL connections, and data encryption are used to strictly control secure access.
- High compatibility: TaurusDB is fully compatible with MySQL. You can easily migrate your MySQL databases to TaurusDB without refactoring existing applications.

- Mass storage: Based on Data Function Virtualization (DFV) distributed storage, TaurusDB supports up to 128 TB of storage.

Service List

- Virtual Private Cloud (VPC)
- Elastic Cloud Server (ECS)
- TaurusDB
- Data Replication Service (DRS)

Notes on Usage

The resources and test data in this practice are for demonstration only. Adjust them as needed.

For more information about TaurusDB data migration, see **From MySQL to TaurusDB**.

1.2 Resource Planning

Table 1-1 Resource planning

Category	Subcategory	Planned Value	Remarks
VPC	VPC name	vpc-mysql	Customize a name for easy identification.
	Region	EU-Dublin	Currently, only Europe-Dublin is supported.
	AZ	AZ3	-
	Subnet	10.0.0.0/24	Select a subnet with sufficient network resources.
	Subnet name	subnet-mysql	Customize a name for easy identification.
ECS (MySQL server)	ECS name	ecs-mysql	Customize a name for easy identification.
	Specifications	s6.xlarge.2 4 vCPUs | 8 GiB	Select specification based on service requirements.
	OS	CentOS 7.6 64	-
	System disk	General purpose SSD 40 GiB	-
	Data disk	Ultra-high I/O, 100 GiB	-
	EIP	Auto assign	Buy an EIP because the public network is selected for the migration task.

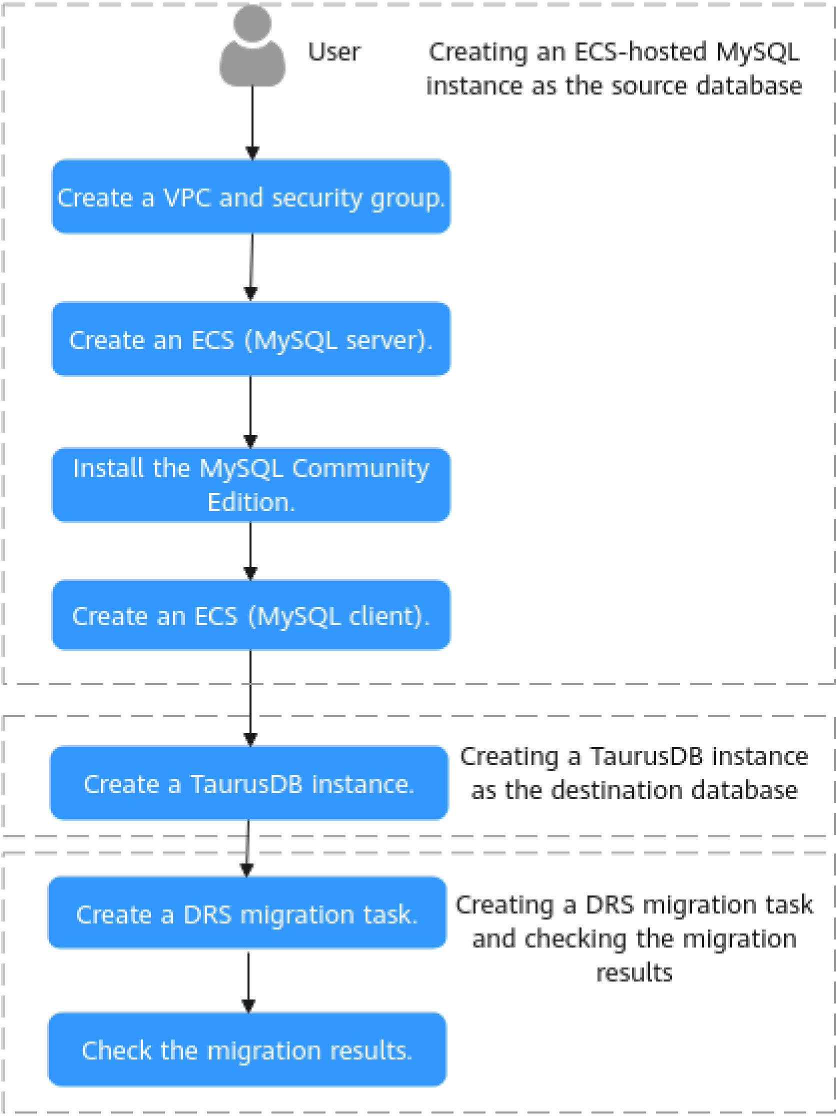
Category	Subcategory	Planned Value	Remarks
ECS (MySQL client)	ECS name	ecs-client	Customize a name for easy identification.
	Specifications	s6.xlarge.2 4 vCPUs | 8 GiB	Select specification based on service requirements.
	OS	CentOS 7.6 64	-
	System disk	General purpose SSD 40 GiB	-
	Data disk	Not required	-
	EIP	Auto assign	Buy an EIP as needed. If you do not need to access the client through a public network, you do not buy an EIP.
TaurusDB	Instance name	gauss-mysql	Customize a name for easy identification.
	DB engine	TaurusDB	-
	DB engine version	MySQL 8.0	-
	AZ type	Single-AZ	-
	AZ	AZ6	-
	Instance specifications	Dedicated Edition	-
	CPU architecture	x86 8 vCPUs | 32 GB	-
DRS migration task	Task name	DRS-TaurusDB	Customize a name.
	Source DB engine	MySQL	In this example, take a MySQL instance (community edition) installed on an ECS as the source database.
	Destination DB engine	TaurusDB	In this example, take a TaurusDB instance as the destination database.
	Network type	Public	In this example, select the public network.

1.3 Operation Process

The process involves creating an ECS, installing a MySQL database on it, buying a TaurusDB instance, and migrating data from the database to the TaurusDB instance.

Figure 1-3 shows the main operation flowchart.

Figure 1-3 Flowchart


User
Creating an ECS-hosted MySQL instance as the source database
Create a VPC and security group.
Create an ECS (MySQL server).
Install the MySQL Community Edition.
Create an ECS (MySQL client).
Create a TaurusDB instance.
Creating a TaurusDB instance as the destination database
Create a DRS migration task.
Creating a DRS migration task and checking the migration results
Check the migration results.

1.4 Procedure

1.4.1 ECS-hosted MySQL Server

1.4.1.1 Creating a VPC and Security Group

This section describes how to create a VPC and security group for your MySQL server and TaurusDB instance.

Creating a VPC

Step 1 Log in to the **management console**.

Step 2 Click in the upper left corner of the management console and select **EU-Dublin**.

Step 3 Click ☰ in the upper left corner of the page and choose **Network > Virtual Private Cloud VPC**.

The VPC console is displayed.

Step 4 On the displayed page, click **Create VPC** in the upper right corner.

Step 5 Configure required parameters.

Step 6 Click **Create Now**.

Step 7 Return to the VPC list and check whether the VPC is created.

If the VPC status becomes available, the VPC has been created.

----End

Creating a Security Group

Step 1 Log in to the **management console**.

Step 2 Click ⚲ in the upper left corner of the management console and select **EU-Dublin**.

Step 3 Click ☰ in the upper left corner of the page and choose **Network > Virtual Private Cloud VPC**.

The VPC console is displayed.

Step 4 In the navigation pane on the left, choose **Access Control** > **Security Groups**.

Step 5 Click **Create Security Group** in the upper right corner of the page.

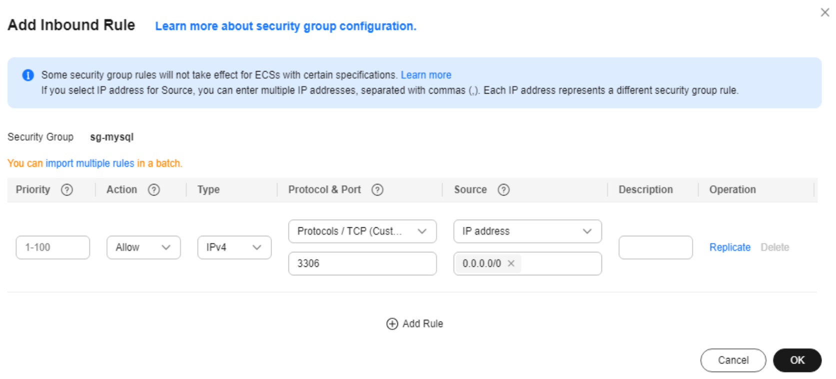
Step 6 In the displayed dialog box, configure parameters as needed.

Step 7 Click **OK**.

Step 8 Return to the security group list, locate the security group **sg-mysql**, and click its name.

Step 9 Click the **Inbound Rules** tab, and then click **Add Rule**.

Step 10 Configure an inbound rule to allow access from database port **3306**.



Add Inbound Rule Learn more about security group configuration.

Some security group rules will not take effect for ECSs with certain specifications. Learn more
If you select IP address for Source, you can enter multiple IP addresses, separated with commas (,). Each IP address represents a different security group rule.

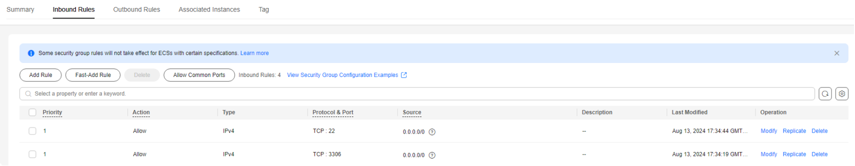
Security Group sg-mysql

You can import multiple rules in a batch.

Priority	Action	Type	Protocol & Port	Source	Description	Operation
1-100	Allow	IPv4	Protocols / TCP (Cust... 3306	IP address 0.0.0.0/0		Replicate Delete

Step 11 Perform **Step 9** to **Step 10** to allow access from database port **22**.

After the rules were configured, the figure similar to the following is displayed.



Some security group rules will not take effect for ECSs with certain specifications. Learn more

Inbound Rules: 4

Priority	Action	Type	Protocol & Port	Source	Description	Last Modified	Operation
1	Allow	IPv4	TCP : 22	0.0.0.0/0	--	Aug 13, 2024 17:34:44 GMT...	Modify Replicate Delete
1	Allow	IPv4	TCP : 3306	0.0.0.0/0	--	Aug 13, 2024 17:34:19 GMT...	Modify Replicate Delete

----End

1.4.1.2 Creating an ECS (MySQL Server)

This section describes how to buy an ECS for installing a MySQL database (community edition).

Step 1 Log in to the **management console**.

Step 2 Click in the upper left corner of the management console and select **EU-Dublin**.

Step 3 Click ☰ in the upper left corner of the page and choose **Compute** > **Elastic Cloud Server**.

Step 4 Click **Buy ECS**.

Step 5 Configure ECS parameters.

1. Set **Specifications** to **General computing** and select **s6.xlarge.2** with 4 vCPUs and 8 GiB.
2. Select the image and disk specifications.

Step 6 Click **Next: Configure Network**.

1. Select the VPC and security group created in **Creating a VPC and Security Group**.
2. Set **EIP** to **Auto assign**, **Billed by** to **Traffic**, and **Bandwidth Size** to **20**. The bandwidth size can be changed as required.

Step 7 Click **Next: Configure Advanced Settings**.

Specify **ECS Name** and **Password**.

Step 8 Click **Next: Confirm**.

Step 9 Select an enterprise project and click **Submit**.

Step 10 Return to the ECS list page and view the creation progress.

When the ECS status changes to **Running**, the ECS has been created.

----End

1.4.1.3 Installing a MySQL Database (Community Edition)

This section describes how to initialize disks and install a MySQL database (community edition).

Log In to the ECS

Step 1 Log in to the **management console**.

Step 2 Click 📍 in the upper left corner of the management console and select **EU-Dublin**.

Step 3 Click ☰ in the upper left corner of the page and choose **Compute** > **Elastic Cloud Server**.

Step 4 Locate the ECS **ecs-mysql** and click **Remote Login** in the **Operation** column.

Step 5 Select **CloudShell-based Login**.

Step 6 Enter the password of user **root**.

NOTE

The password is the one you specified during the ECS creation.

----End

Initializing Disks

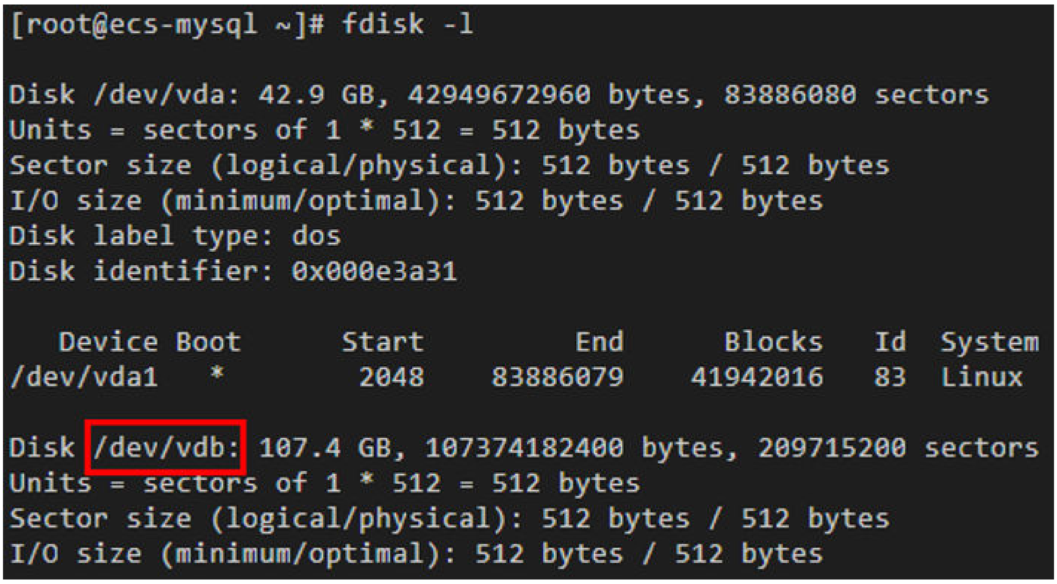
Step 1 Create the **mysql** folder.

mkdir /mysql

Step 2 View data disk information.

fdisk -l

The command output is as follows.



```
[root@ecs-mysql ~]# fdisk -l

Disk /dev/vda: 42.9 GB, 42949672960 bytes, 83886080 sectors
Units = sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disk label type: dos
Disk identifier: 0x000e3a31

   Device Boot      Start         End      Blocks   Id  System
/dev/vda1   *        2048    83886079    41942016   83  Linux

Disk /dev/vdb: 107.4 GB, 107374182400 bytes, 209715200 sectors
Units = sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
```

Step 3 Initialize the data disk.

mkfs.ext4 /dev/vdb

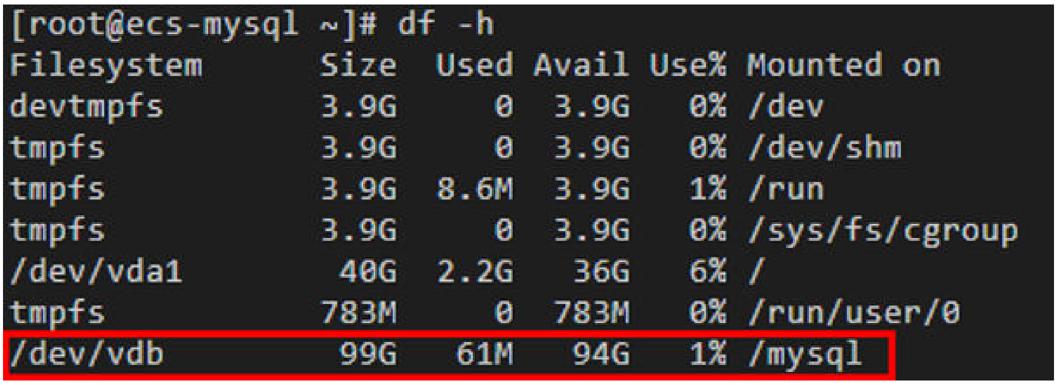
Step 4 Attach the disk.

mount /dev/vdb /mysql

Step 5 Check whether the disk has been attached.

df -h

If the following output is returned, the disk has been attached.



```
[root@ecs-mysql ~]# df -h
Filesystem      Size  Used Avail Use% Mounted on
devtmpfs        3.9G     0  3.9G   0% /dev
tmpfs           3.9G     0  3.9G   0% /dev/shm
tmpfs           3.9G  8.6M  3.9G   1% /run
tmpfs           3.9G     0  3.9G   0% /sys/fs/cgroup
/dev/vda1        40G  2.2G   36G   6% /
tmpfs           783M     0  783M   0% /run/user/0
/dev/vdb         99G   61M   94G   1% /mysql
```

Step 6 Create a folder and switch to the **install** folder.

mkdir -p /mysql/install/data

mkdir -p /mysql/install/tmp

mkdir -p /mysql/install/file

mkdir -p /mysql/install/log

cd /mysql/install

Step 7 Download and install **the MySQL client**.

Step 8 Initialize the MySQL client.

/mysql/install/mysql-8.0.22/bin/mysqld --defaults-file= /etc/my.cnf --initialize-insecure

Step 9 Start the MySQL client.

nohup /mysql/install/mysql-8.0.22/bin/mysqld --defaults-file= /etc/my.cnf &

Step 10 Connect to the MySQL client.

/mysql/install/mysql-8.0.22/bin/mysql

Step 11 Create user **root** and assign the required permissions to it.

grant all privileges on *.* to 'root'@'%' identified by 'xxx' with grant option;FLUSH PRIVILEGES;

----End

1.4.1.4 Creating an ECS and Installing a MySQL Client on It

Step 1 Create an ECS for installing a MySQL client by referring to **Creating an ECS (MySQL Server)**.

NOTE

- This ECS must be in the same region, AZ, VPC, and security group as the ECS where the MySQL server is deployed.
- Data disks are not required.
- This ECS name is **ecs-client**.
- Other parameters are the same as those of the ECS where the MySQL server is deployed.

Step 2 Download and install the MySQL client. For details, see **How Can I Install the MySQL Client?**

----End

1.4.2 Cloud Migration

This chapter describes how to create a TaurusDB instance, create a DRS migration task, and migrate data from the ECS-hosted MySQL server to the TaurusDB instance.

1.4.2.1 Creating a TaurusDB Instance

This section describes how to create a TaurusDB instance that is in the same VPC and security group as the ECS-hosted MySQL server.

Step 1 Log in to the **management console**.

Step 2 Click 📍 in the upper left corner of the management console and select **EU-Dublin**.

Step 3 Click ☰ in the upper left corner of the page and choose **Databases** > **TaurusDB**.

Step 4 In the upper right corner, click **Buy DB Instance**.

Step 5 Configure the instance name and basic information.

Step 6 Configure instance specifications.

Step 7 Select the VPC and security group.

The VPC and security group have been created in **Creating a VPC and Security Group**.

Step 8 Configure the instance password.

Step 9 Configure an enterprise project.

Step 10 Click **Next**.

Step 11 After confirming the settings, click **Submit**.

Step 12 Return to the instance list.

If the instance status becomes **Available**, the instance has been created.

----End

1.4.2.2 Creating a DRS Migration Task

This section describes how to create a DRS migration task to migrate the **sbtest** database from the ECS-hosted MySQL server to the TaurusDB instance.

Step 1 Log in to the **management console**.

Step 2 Click 📍 in the upper left corner of the management console and select **EU-Dublin**.

Step 3 Click ☰ in the upper left corner of the page and choose **Database** > **Data Replication Service**.

Step 4 In the upper right corner, click **Create Migration Task**.

Step 5 Configure parameters as needed.

1. Specify a task name.
2. Configure replication instance details as needed.

   Set **Destination DB Instance** to the TaurusDB instance created in **Creating a TaurusDB Instance**.
3. Set **Enterprise Project** to **default**.

Step 6 Click **Create Now**.

It takes about 5 to 10 minutes to create a replication instance.

Step 7 Configure source and destination database information.

1. Configure source database information.
2. Click **Test Connection**.

   If a successful connection message is returned, you have logged in to the source database.
3. Configure the username and password for the destination database.
4. Click **Test Connection**.

   If a successful connection message is returned, you have logged in to the destination database.

Step 8 Click **Next**.

Step 9 Confirm the users, snapshots, and migration objects to be migrated.

Set **Migrate Object** to **All**.

Step 10 Click **Next**.

Step 11 View pre-check results.

Step 12 If the check is complete and the check success rate is 100%, click **Next**.

Step 13 Click **Submit**.

Return to the **Online Migration Management** page and check the migration task status.

It takes several minutes to complete.

If the status changes to **Completed**, the migration task has been created.

----End

1.4.2.3 Checking the Migration Results

You can check the migration results with either of the following methods:

Method 1: (Automatically) **Check the migration results on the DRS console**. DRS can compare migration objects, users, and data of source and destination databases and obtain the migration results.

Method 2: (Manually) **Check the migration results on the TaurusDB console**. Log in to the destination database to check whether the databases, tables, and data are migrated. Manually confirm the data migration status.

Checking the Migration Results on the DRS Console

Step 1 Log in to the **management console**.

Step 2 Click 📍 in the upper left corner of the management console and select **EU-Dublin**.

Step 3 Click ☰ in the upper left corner of the page and choose **Databases** > **Data Replication Service**.

Step 4 Click the DRS instance name.

Step 5 Click **Migration Comparison**.

Step 6 Under the **Compare Data - Validate ALL Rows/Values** and **Compare Data - Double Check During Cutover** tabs, check whether the objects of the source database have been migrated to destination database.

----End

Checking the Migration Results on the TaurusDB Console

Step 1 Log in to the **management console**.

Step 2 Click 📍 in the upper left corner of the management console and select **EU-Dublin**.

Step 3 Click ☰ in the upper left corner of the page and choose **Databases** > **TaurusDB**.

Step 4 Locate the required TaurusDB instance and choose **More** > **Log In** in the **Operation** column.

Step 5 In the displayed dialog box, enter the password and click **Test Connection**.

Step 6 After the connection test is successful, click **Log In**.

Step 7 Check and confirm the destination database name and table name. Check whether the data migration is complete.

----End

Testing TaurusDB Performance

After the migration is complete, test TaurusDB performance by referring to **Performance White Paper**.

2 From Other Cloud MySQL to TaurusDB

2.1 Overview

Description

This section includes the following content:

- Create a TaurusDB instance.
- Migrate data from MySQL on other clouds to TaurusDB.

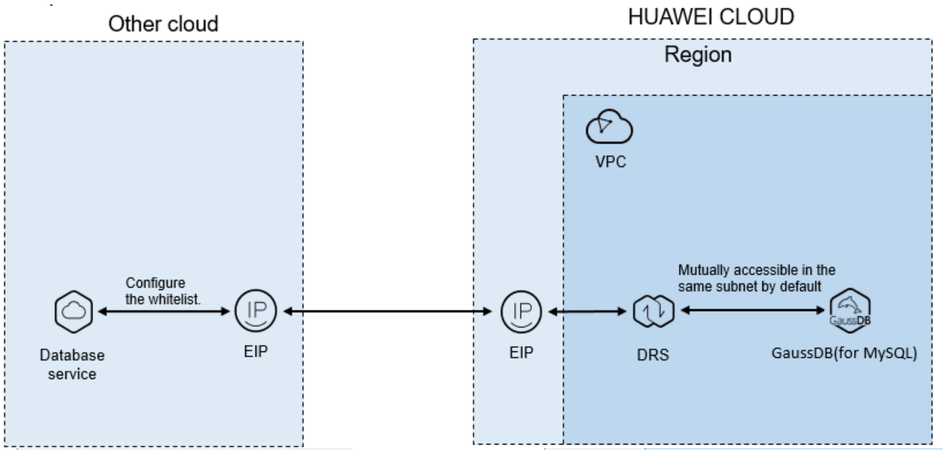
Prerequisites

- You have registered with Huawei Cloud.
- Your account balance is greater than or equal to $0 USD.

Deployment Architecture

In this example, the source is a MySQL database on other cloud platforms and the destination is a Huawei Cloud TaurusDB instance. Data is migrated from the source to the destination over a public network. For details about the deployment architecture, see **Figure 2-1**.

Figure 2-1 Deployment architecture



Service List

- Virtual Private Cloud (VPC)
- TaurusDB
- Data Replication Service (DRS)

Before You Start

- The resource planning in this best practice is for demonstration only. Adjust it as needed.
- All settings in this best practice are for reference only. For more information about MySQL migration, see **From MySQL to GaussDB(for MySQL) Primary/Standby**.

2.2 Resource Planning

Table 2-1 Resource planning

Category	Subcategory	Plan	Description
VPC	VPC name	vpc-DRStest	Specify a name that is easy to identify.
	Region	EU-Dublin	To achieve lower network latency, select the region nearest to you.
	AZ	AZ 1	-
	Subnet	10.0.0.0/24	Select a subnet with sufficient network resources.
	Subnet name	subnet-drs01	Specify a name that is easy to identify.
Other cloud MySQL	DB engine version	MySQL 5.7	-
	IP address	10.154.217.42	Enter an IP address.
	Port	3306	-
TaurusDB instance	Instance name	gauss-drstar	Specify a name that is easy to identify.
	DB engine version	MySQL 8.0	-
	AZ type	Single AZ	In this example, a single AZ is used. To improve service reliability, select multiple AZs.

Category	Subcategory	Plan	Description
	AZ	AZ1	AZ1 is selected in this example. To improve service reliability, deploy the instance across multiple AZs.
	Instance class	Dedicated 4 vCPUs | 16 GB	-
DRS migration task	Task name	DRS-test-migrate	Specify a name that is easy to identify.
	Source DB engine	MySQL	-
	Destination DB engine	TaurusDB	-
	Network type	Public network	Public network is used in this example.

2.3 Creating a VPC and Security Group

Create a VPC and security group for a TaurusDB instance.

Creating a VPC

Step 1 Log in to the **management console**.

Step 2 Click 📍 in the upper left corner of the management console and select region EU-Dublin.

Step 3 Under the service list, choose **Networking** > **Virtual Private Cloud**.

Step 4 Click **Create VPC**.

Step 5 Configure parameters as needed and click **Create Now**.

Step 6 Return to the VPC list and check whether the VPC is created.

If the VPC status becomes available, the VPC has been created.

----End

Creating a Security Group

Step 1 Log in to the **management console**.

Step 2 Click 📍 in the upper left corner of the management console and select region EU-Dublin.

Step 3 Under the service list, choose **Networking** > **Virtual Private Cloud**.

Step 4 In the navigation pane, choose **Access Control** > **Security Groups**.

Step 5 Click **Create Security Group**.

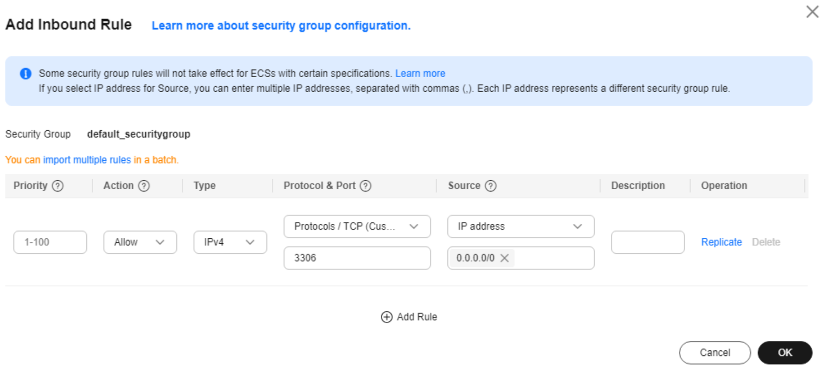
Step 6 Configure parameters as needed.

Step 7 Click **OK**.

Step 8 Return to the security group list and click the security group name (**sg-DRS01** in this example).

Step 9 Click the **Inbound Rules** tab, and then click **Add Rule**.

Step 10 Configure an inbound rule to allow access from database port **3306**.



Add Inbound Rule Learn more about security group configuration.

Some security group rules will not take effect for ECSs with certain specifications. Learn more
If you select IP address for Source, you can enter multiple IP addresses, separated with commas (,). Each IP address represents a different security group rule.

Security Group default_securitygroup

You can import multiple rules in a batch.

Priority	Action	Type	Protocol & Port	Source	Description	Operation
1-100	Allow	IPv4	Protocols / TCP (Cus... 3306	IP address 0.0.0.0/0		Replicate Delete

----End

2.4 Creating a TaurusDB Instance

This section describes how to create a Huawei Cloud TaurusDB instance.

Step 1 Log in to the **management console**.

Step 2 Click in the upper left corner of the management console and select region EU-Dublin.

Step 3 Under the service list, choose **Databases** > **TaurusDB**.

Step 4 On the **Instances** page, click **Buy DB Instance**.

Step 5 Configure the instance name and basic information.

Step 6 Configure instance specifications.

Step 7 Select a VPC and security group for the instance and configure the database port.

The VPC and security group have been created in **Creating a VPC and Security Group**.

Step 8 Configure the instance password.

Step 9 Configure an enterprise project.

Step 10 Click **Next**. If you do not need to modify your settings, click **Submit**.

Step 11 Return to the instance list. If the instance becomes **Available**, the instance has been created.

----End

2.5 Configuring a MySQL Instance on Other Clouds

Prerequisites

- You have purchased a MySQL instance on other platforms.
- The MySQL account has the migration permissions listed in **Permission Requirements**.

Permission Requirements

To migrate data from a MySQL database on other clouds to a TaurusDB instance, the following permissions are required.

Table 2-2 Required permissions

Database	Full Migration Permission	Full+Incremental Migration Permission
Source DB (MySQL)	SELECT, SHOW VIEW, and EVENT	SELECT, SHOW VIEW, EVENT, LOCK TABLES, REPLICATION SLAVE, and REPLICATION CLIENT

For details about MySQL authorization operations, see **operation guide**.

Network Configuration

Enable public accessibility for the source database. The method for enabling public accessibility depends on the cloud database vendor. For details, see the official documents of the corresponding cloud database vendor.

2.6 Creating a DRS Migration Task

This section describes how to create a DRS instance and migrate data from a MySQL database on other clouds to a TaurusDB instance.

Pre-migration Check

Before creating a migration task, check the migration environment.

This section describes how to migrate data from a MySQL database to TaurusDB. For details, see **Before You Start**.

Creating a Migration Task

Step 1 Log in to the **management console**.

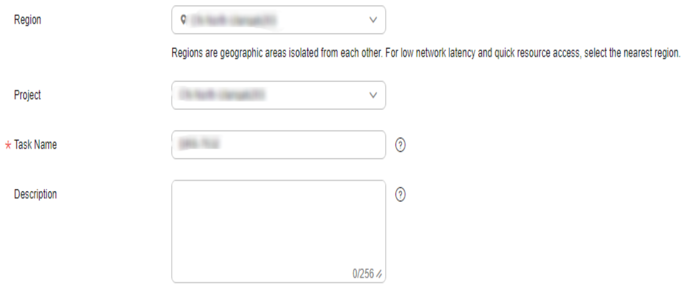
Step 2 Click in the upper left corner of the management console and select region EU-Dublin.

Step 3 Under the service list, choose **Databases** > **Data Replication Service**.

Step 4 In the upper right corner, click **Create Migration Task**.

Step 5 Configure parameters as needed.

1. Specify a migration task name.



Region

Regions are geographic areas isolated from each other. For low network latency and quick resource access, select the nearest region.

Project

* Task Name

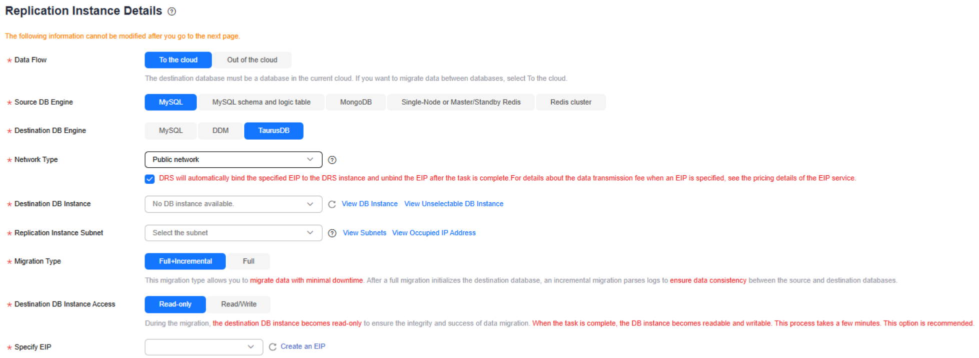
Description

0/256

2. Configure replication instance details as needed.

   Select the TaurusDB instance created in **Creating a TaurusDB Instance** as the destination database.

Figure 2-2 Replication instance information



Replication Instance Details

The following information cannot be modified after you go to the next page.

* Data Flow: [✓ To the cloud] Out of the cloud

The destination database must be a database in the current cloud. If you want to migrate data between databases, select To the cloud.

* Source DB Engine: [✓ MySQL] MySQL schema and logic table MongoDB Single-Node or Master/Standby Redis Redis cluster

* Destination DB Engine: MySQL DDM [✓ TaurusDB]

* Network Type: Public network

☑ DRS will automatically bind the specified EIP to the DRS instance and unbind the EIP after the task is complete.For details about the data transmission fee when an EIP is specified, see the pricing details of the EIP service.

* Destination DB Instance: No DB instance available. View DB Instance View Unselectable DB Instance

* Replication Instance Subnet: Select the subnet View Subnets View Occupied IP Address

* Migration Type: [✓ Full+Incremental] Full

This migration type allows you to migrate data with minimal downtime. After a full migration initializes the destination database, an incremental migration parses logs to ensure data consistency between the source and destination databases.

* Destination DB Instance Access: [✓ Read-only] Read/Write

During the migration, the destination DB instance becomes read-only to ensure the integrity and success of data migration. When the task is complete, the DB instance becomes readable and writable. This process takes a few minutes. This option is recommended.

* Specify EIP: Create an EIP

Step 6 Click **Create Now**.

It takes about 5 to 10 minutes to create a replication instance.

Step 7 Configure a whitelist for the source database to manage network access.

Add the EIP of the DRS replication instance to the whitelist of the source MySQL database to ensure that the source database can communicate with the DRS instance.

The method for configuring the whitelist depends on the cloud database vendor. For details, see the official documents of the corresponding cloud database vendor.

Step 8 Configure source and destination database information.

1. Configure the source database information and click **Test Connection**. If a successful test message is returned, login to the destination is successful.
2. Configure destination database information and click **Test Connection**. If a successful test message is returned, login to the destination is successful.

Step 9 Click **Next**.

Step 10 On the **Set Task** page, select migration accounts and objects.

- Select **No** for **Migrate Account**.
- Select **All** for **Migrate Object**.

Step 11 Click **Next**. On the **Check Task** page, check the migration task.

- If any check fails, review the cause and rectify the fault. After the fault is rectified, click **Check Again**.
- If all check items are successful, click **Next**.

Step 12 Click **Submit**.

Return to the **Online Migration Management** page and check the migration task status.

It takes several minutes to complete.

If the status changes to **Completed**, the migration task has been created.

NOTE

- Currently, MySQL to TaurusDB migration supports two modes: full migration and full +incremental migration.
- If you create a full migration task, the task automatically stops after the full data is migrated to the destination.
- If you create a full+incremental migration task, a full migration is executed first. After the full migration is complete, an incremental migration starts.
- During the incremental migration, data is continuously migrated so the task will not automatically stop.

----End

2.7 Checking Migration Results

You can use either of the following methods to check the migration results:

1. DRS compares migration objects, users, and data and provide comparison results. For details, see **Checking the Migration Results on the DRS Console**.
2. Log in to the destination side to check whether the databases, tables, and data are migrated. Confirm the data migration status. For details, see **Checking the Migration Results on the TaurusDB Console**.

Checking the Migration Results on the DRS Console

Step 1 Log in to the **management console**.

Step 2 Click ⊙ in the upper left corner of the management console and select region EU-Dublin.

Step 3 Under the service list, choose **Databases** > **Data Replication Service**.

Step 4 Click the DRS instance name.

Step 5 Choose **Migration Comparison** and select **Object-Level Comparison** to check whether database objects are missing.

Step 6 Click **Data-Level Comparison** and check whether the number of rows of migrated objects is consistent.

Step 7 Click **Account-Level Comparison** and check whether the accounts and permissions of the source and destination instances are the same.

----End

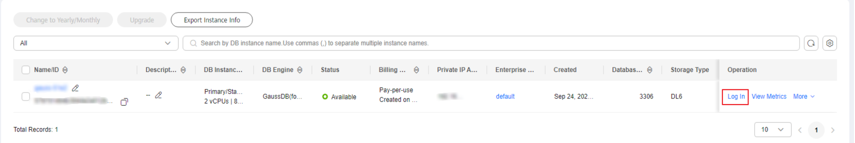
Checking the Migration Results on the TaurusDB Console

Step 1 Log in to the **management console**.

Step 2 Click ⊙ in the upper left corner of the management console and select region EU-Dublin.

Step 3 Under the service list, choose **Databases** > **TaurusDB**.

Step 4 On the **Instances** page, locate the destination instance, and click **Log In** in the **Operation** column.



Change to Yearly/Monthly
Upgrade
Export Instance Info
All
Search by DB instance name.Use commas (,) to separate multiple instance names.

□	Name/ID	Descript...	DB Instanc...	DB Engine	Status	Billing ...	Private IP A...	Enterprise ...	Created	Databas...	Storage Type	Operation
□	[redacted]	--	Primary/Sta... 2 vCPUs | 8...	GaussDB(fo...	○ Available	Pay-per-use Created on ...	[redacted]	default	Sep 24, 202...	3306	DL6	Log In View Metrics More

Total Records: 1

Step 5 In the dialog box that is displayed, enter the password and click **Test Connection**.

Step 6 After the connection is successful, click **Log In**.

Step 7 Check whether the destination databases and tables are the same as the source instance. Check whether migration is complete.

----End

3 Enabling Read/Write Splitting

3.1 User Authentication

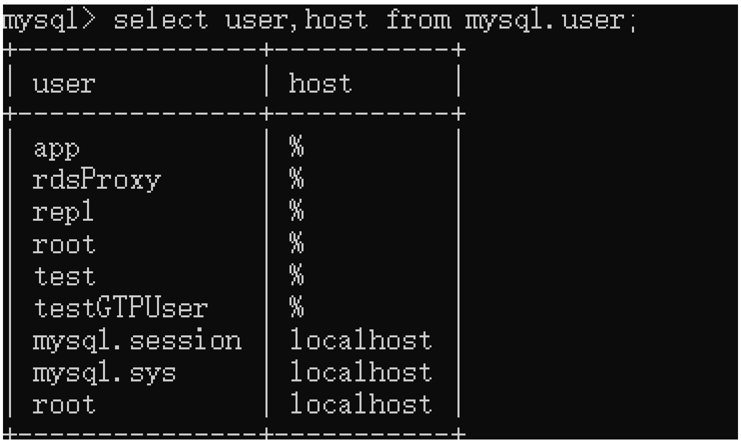
You must have the remote login permission before using a database proxy to log in to databases.

Procedure

Step 1 Connect to a TaurusDB instance.

Step 2 Check whether the host of the used account contains a proxy address.

```
SELECT user,host FROM mysql.user;
```



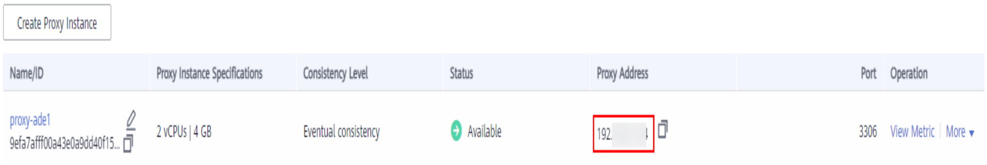
```
mysql> select user,host from mysql.user;
+---------------+-----------+
| user          | host      |
+---------------+-----------+
| app           | %         |
| rdsProxy      | %         |
| repl          | %         |
| root          | %         |
| test          | %         |
| testGTPUser   | %         |
| mysql.session | localhost |
| mysql.sys     | localhost |
| root          | localhost |
+---------------+-----------+
```

Obtaining a proxy address:

1. Log in to the management console.
2. Click 📍 in the upper left corner and select a region and a project.
3. Click ☰ in the upper left corner of the page, choose **Database** > **TaurusDB**.
4. On the **Instances** page, click the instance name.
5. In the navigation pane, choose **Database Proxy**. Obtain the proxy address in either of the following ways:

Method 1: In the proxy instance list, locate the proxy instance and view the value in the **Proxy Address** column.

Figure 3-1 Viewing the proxy address in the proxy instance list page

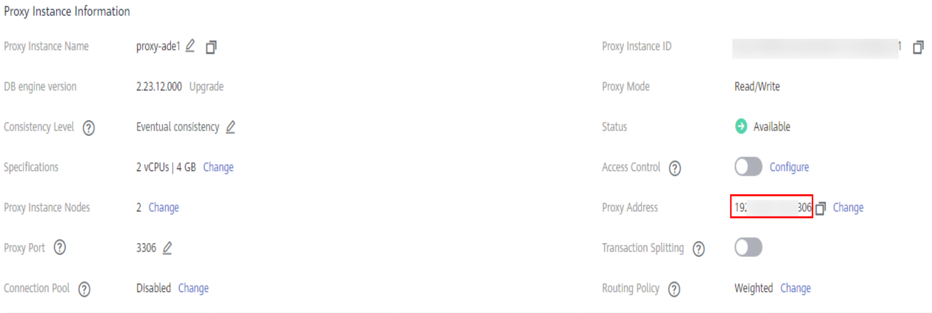


Create Proxy Instance

Name/ID	Proxy Instance Specifications	Consistency Level	Status	Proxy Address	Port	Operation
proxy-ade1 9efa7afff00a43e0a9dd40f15...	2 vCPUs | 4 GB	Eventual consistency	Available	192.[redacted]	3306	View Metric | More ▾

Method 2: In the proxy instance list, click the name of the proxy instance to go to the **Basic Information** page. In the **Proxy Instance Information** area, view the **Proxy Address** field.

Figure 3-2 Viewing the proxy address in the proxy instance information page



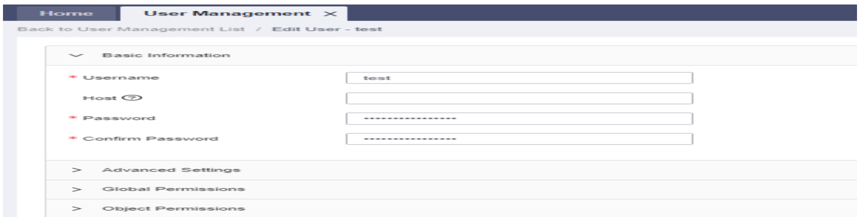
Proxy Instance Information

Proxy Instance Name	proxy-ade1	Proxy Instance ID	[redacted]1
DB engine version	2.23.12.000 Upgrade	Proxy Mode	Read/Write
Consistency Level	Eventual consistency	Status	Available
Specifications	2 vCPUs | 4 GB Change	Access Control	Configure
Proxy Instance Nodes	2 Change	Proxy Address	19[redacted]306 Change
Proxy Port	3306	Transaction Splitting	
Connection Pool	Disabled Change	Routing Policy	Weighted Change

Step 3 If the host does not contain the CIDR block where the proxy instance is located, assign the remote access permission to the host.

For example, allowing user **root** to access the TaurusDB server from the IP address range starting with 192.168.0.

Alternatively, set **Host** on the **User Management** page of the DAS console. For details, see .



Home | User Management ×

Back to User Management List / Edit User - test

Basic Information

* Username	test
Host	
* Password	-----------------
* Confirm Password	-----------------

> Advanced Settings

> Global Permissions

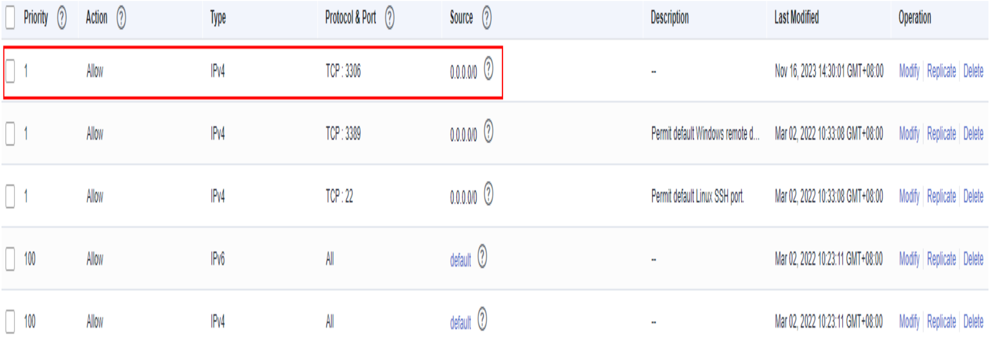
> Object Permissions

Step 4 When modifying a security group, ensure that the inbound and outbound rules allow access of the proxy address. The default port is **3306**.

1. Log in to the management console.
2. Click in the upper left corner and select a region and a project.
3. Click ☰ in the upper left corner of the page, choose **Database** > **TaurusDB**.
4. On the **Instances** page, click the instance name to go to the **Basic Information** page.

5. In the **Network Information** area, click the security group name.
6. On the **Inbound Rules** tab, check whether access through port **3306** is allowed by default.

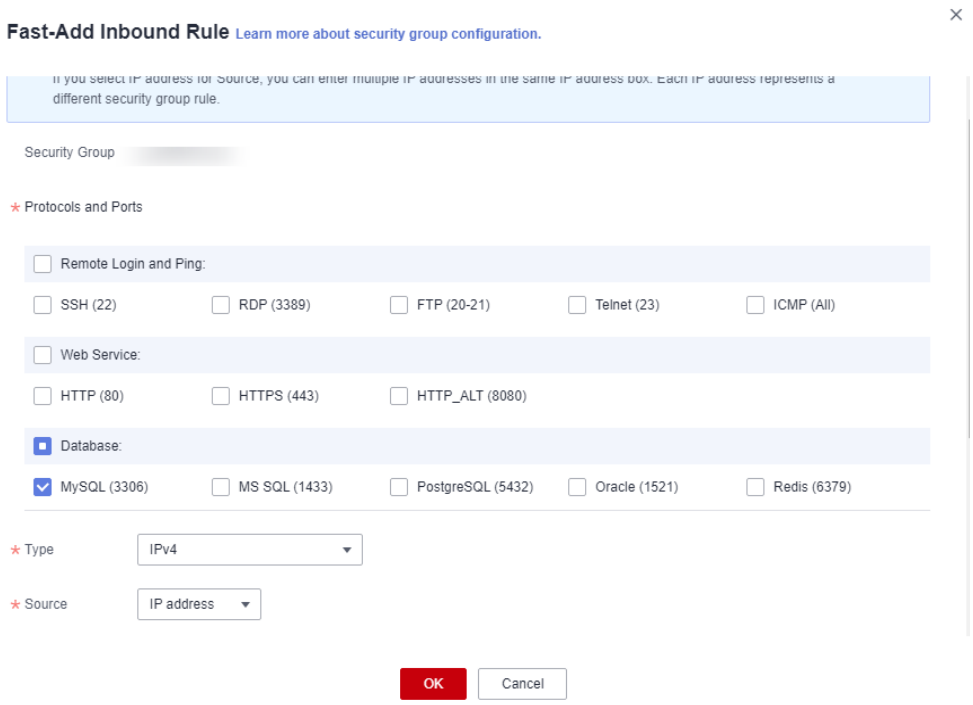
Figure 3-3 Allowing access through port 3306



□ Priority ⓘ	Action ⓘ	Type	Protocol & Port ⓘ	Source ⓘ	Description	Last Modified	Operation
□ 1	Allow	IPv4	TCP : 3306	0.0.0.0/0 ⓘ	--	Nov 16, 2023 14:30:01 GMT+08:00	Modify | Replicate | Delete
□ 1	Allow	IPv4	TCP : 3389	0.0.0.0/0 ⓘ	Permit default Windows remote d...	Mar 02, 2022 10:33:08 GMT+08:00	Modify | Replicate | Delete
□ 1	Allow	IPv4	TCP : 22	0.0.0.0/0 ⓘ	Permit default Linux SSH port.	Mar 02, 2022 10:33:08 GMT+08:00	Modify | Replicate | Delete
□ 100	Allow	IPv6	All	default ⓘ	--	Mar 02, 2022 10:23:11 GMT+08:00	Modify | Replicate | Delete
□ 100	Allow	IPv4	All	default ⓘ	--	Mar 02, 2022 10:23:11 GMT+08:00	Modify | Replicate | Delete

If this rule does not exist, click **Fast-Add Rule**. In the displayed dialog box, select **MySQL (3306)** and click **OK**.

Figure 3-4 Fast adding port 3306



Fast-Add Inbound Rule Learn more about security group configuration.

If you select IP address for Source, you can enter multiple IP addresses in the same IP address box. Each IP address represents a different security group rule.

Security Group [redacted]

* Protocols and Ports

□ Remote Login and Ping:
□ SSH (22) □ RDP (3389) □ FTP (20-21) □ Telnet (23) □ ICMP (All)

□ Web Service:
□ HTTP (80) □ HTTPS (443) □ HTTP_ALT (8080)

■ Database:
☑ MySQL (3306) □ MS SQL (1433) □ PostgreSQL (5432) □ Oracle (1521) □ Redis (6379)

* Type: IPv4

* Source: IP address

NOTE

When you use the MySQL 8.0 client to access the read/write splitting of the database, the error message "auth user failed" may be displayed.

Add **--default-auth=mysql_native_password** when connecting to the database.

----End

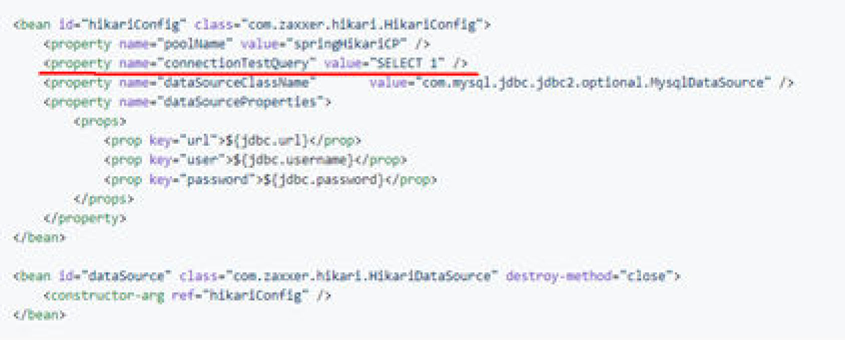
3.2 Connection Pool Configuration

When the connection pool is used, you need to configure the following parameters to ensure that some connections will not be used even though they are disconnected due to timeout.

- For JDBC connection pool and Druid connection pool:

  testOnBorrow = **true**
- For HikariCP connection pool:

  connectionTestQuery = **SELECT 1**



```
<bean id="hikariConfig" class="com.zaxxer.hikari.HikariConfig">
    <property name="poolName" value="springHikariCP" />
    <property name="connectionTestQuery" value="SELECT 1" />
    <property name="dataSourceClassName"       value="com.mysql.jdbc.jdbc2.optional.MysqlDataSource" />
    <property name="dataSourceProperties">
        <props>
            <prop key="url">${jdbc.url}</prop>
            <prop key="user">${jdbc.username}</prop>
            <prop key="password">${jdbc.password}</prop>
        </props>
    </property>
</bean>

<bean id="dataSource" class="com.zaxxer.hikari.HikariDataSource" destroy-method="close">
    <constructor-arg ref="hikariConfig" />
</bean>
```

3.3 Routing Read Requests to the Primary Node

- If there are SELECT statements in transactions, the transaction requests are routed to the primary node. If **SET AUTOCOMMIT=0** is added before a SELECT statement, the transaction requests are routed to the primary node.
- If all read replicas are abnormal or the read weights allocated to the read replicas are 0, requests will be routed to the primary node. You can set read weights allocated to read replicas and primary node after read/write splitting is enabled.
- During the execution of SQL statements:
  - If multi-statements (for example, **insert xxx;select xxx**) are executed, all subsequent requests will be routed to the primary node. To restore the read/write splitting function, disconnect the connection from your applications and establish a connection again.
  - Read operations with locks (for example, **SELECT for UPDATE**) will be routed to the primary node.
  - When **/*FORCE_MASTER*/** is used, requests will be routed to the primary node.
  - If **the HANDLER statement** is executed, all subsequent requests will be routed to the primary node by default. To restore read/write splitting, disconnect the connection and reestablish a connection.