Relational Database Service

Getting Started

Issue 01

Date 2024-12-30

HUAWEI CLOUD COMPUTING TECHNOLOGIES CO., LTD.

Trademarks and Permissions


and other Huawei trademarks are the property of Huawei Technologies Co., Ltd.
All other trademarks and trade names mentioned in this document are the property of their respective holders.

Notice

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

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

Contents

1 Getting Started with RDS for MySQL

1.1 Buying a DB Instance and Connecting to It Using a MySQL Client

After buying a DB instance, you can connect to it using a Linux ECS with a MySQL client installed over a private network. This section describes how to access a DB instance from an ECS using a standard MySQL client.

This section introduces how to connect to a DB instance with SSL disabled. To connect to a DB instance with SSL enabled, see **Using MySQL CLI to Connect to an Instance Through a Private Network**.

Operation Process

Process	Description
Preparations	Sign up for a HUAWEI ID, enable Huawei Cloud services, make sure you have a valid payment method configured, create IAM users, and grant them specific RDS permissions.
Step 1: Buy an RDS for MySQL DB Instance	Select required basic settings and additional options and buy an RDS for MySQL DB instance.
Step 2: Buy an ECS	If you want to connect to a DB instance using a MySQL client, you need to prepare a server, install the MySQL client on the server, and run the connection command. Purchase a Linux ECS that is in the same region and VPC as your DB instance. If you have purchased a Windows ECS, you can connect to the DB instance using MySQL-Front. For details, see **Buying a DB Instance and Connecting to It Using MySQL-Front**.

Process	Description
Step 3: Test Connectivity and Install a MySQL Client	Test the network connectivity between the ECS and the floating IP address and port of the DB instance, and install a MySQL client on the ECS.
Step 4: Connect to the DB Instance Using a CLI (Non-SSL Connection)	Use a command-line interface (CLI) to connect to the DB instance using the floating IP address and port.

Preparations

1. **Sign up for a HUAWEI ID and enable Huawei Cloud services**.
2. Before purchasing DB instances, ensure that your account balance is sufficient. **Top up your account** if required.
3. For fine-grained permissions management on Huawei Cloud resources, use Identity and Access Management (IAM) to create a user or user group and grant it specific operation permissions. For details, see **Creating a User and Granting Permissions**.

Step 1: Buy an RDS for MySQL DB Instance

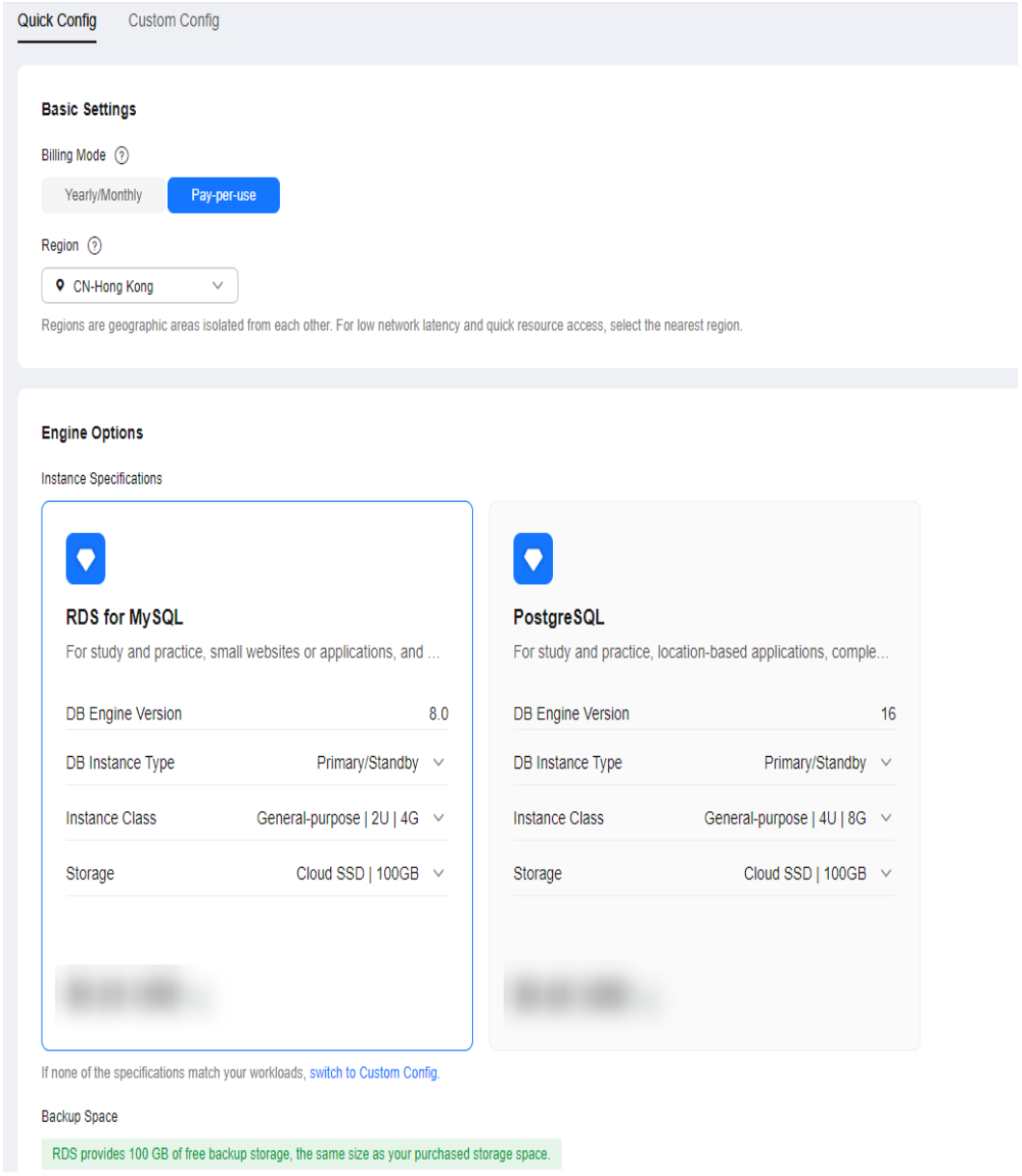
1. Go to the **Buy DB Instance** page.
2. On the **Quick Config** page, set basic parameters.

   NOTE

   Only mandatory parameters are provided on the **Quick Config** page. If the available parameters do not match your workloads, try **Custom Config**.

   The following parameter settings are only for reference.

Figure 1-1 Basic Settings

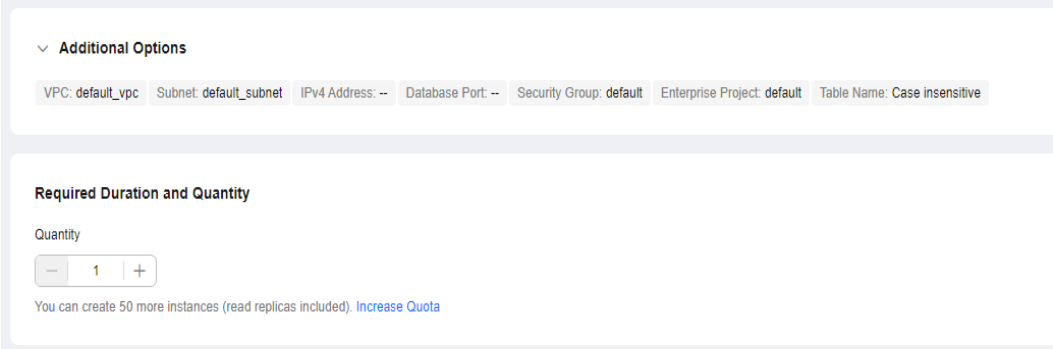


Parameter	Example Value	Description
Billing Mode	Pay-per-use	The billing mode of an instance. • **Yearly/Monthly**: A prepaid billing mode in which you pay for resources before using it. Bills are settled based on the subscription period. The longer the subscription, the bigger the discount. This mode is a good option for long-term, stable services. • **Pay-per-use**: A postpaid billing mode. You pay as you go and just pay for what you use. The DB instance usage is calculated by the second but billed every hour. This mode allows you to adjust resource usage easily. You neither need to prepare for resources in advance, nor end up with excessive or insufficient preset resources.
Region	CN-Hong Kong	The region where your resources are located. **NOTE** Products in different regions cannot communicate with each other through a private network. After a DB instance is created, the region cannot be changed. Therefore, exercise caution when selecting a region.
DB Engine Version	8.0	The database version.
DB Instance Type	Primary/ Standby	The architecture type of an instance. **Primary/Standby**: An HA architecture. In a primary/standby pair, each instance has the same instance class. When a primary instance is being created, a standby instance is provisioned along with it to provide data redundancy. The standby instance is invisible to you after being created.
Instance Class	General-purpose | 2U | 4G	The vCPU and memory of an instance.
Storage	Cloud SSD | 100GB	The storage space of an instance. It contains the system overhead required for inodes, reserved blocks, and database operation.

Parameter	Example Value	Description
Disk Encryption	Disable	Enabling disk encryption improves data security, but slightly affects the read and write performance of the database. If a shared KMS key is used, the corresponding CTS events are **createdatakey** and **decrydatakey**. Only the key owner can receive the events.

3. Complete advanced configurations.

Figure 1-2 Additional Options

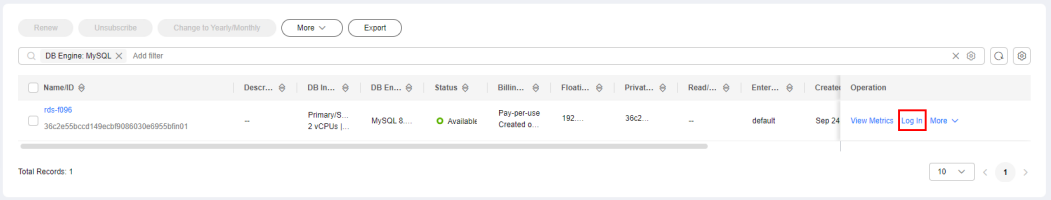


Parameter	Example Value	Description
VPC	vpc-default	The virtual network in which your instance is located. A VPC can isolate networks for different workloads. If no VPC is available, click **Create VPC**. After a VPC is created, click . For details, see **Creating a VPC and Subnet**.
Subnet	subnet-default	A subnet provides dedicated network resources that are logically isolated from other networks for network security.
Security Group	default	It can enhance security by controlling access to RDS for MySQL from other services.
Enterprise Project	default	If your account has been associated with an enterprise project, select the target project from the **Enterprise Project** drop-down list. For more information about enterprise projects, see ***Enterprise Management User Guide***.

Parameter	Example Value	Description
Table Name	Case insensitive	Whether table names are case sensitive. Restoration may fail if the case sensitivity settings of table names on the source and target instances are different. The case sensitivity of table names for created RDS for MySQL 8.0 instances cannot be changed.
Quantity	1	The number of instances to be created in a batch.

4. Click **Buy**.
5. Check the purchased DB instance.

Figure 1-3 Instance successfully purchased



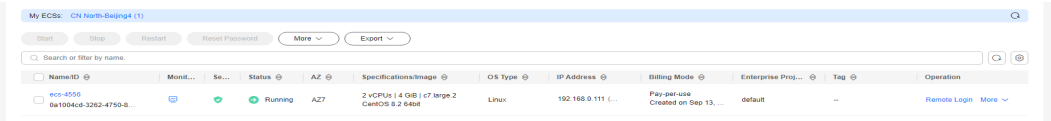
Step 2: Buy an ECS

1. Go to the **Elastic Cloud Server console**.
2. Check whether there is a Linux ECS.
   - If yes, go to **3**.
   - If no, purchase an ECS and select Linux (for example, CentOS) as its OS.

     To download a MySQL client to the ECS, bind an EIP to the ECS. The ECS must be in the same region, VPC, and security group as the RDS for MySQL DB instance for mutual communications.

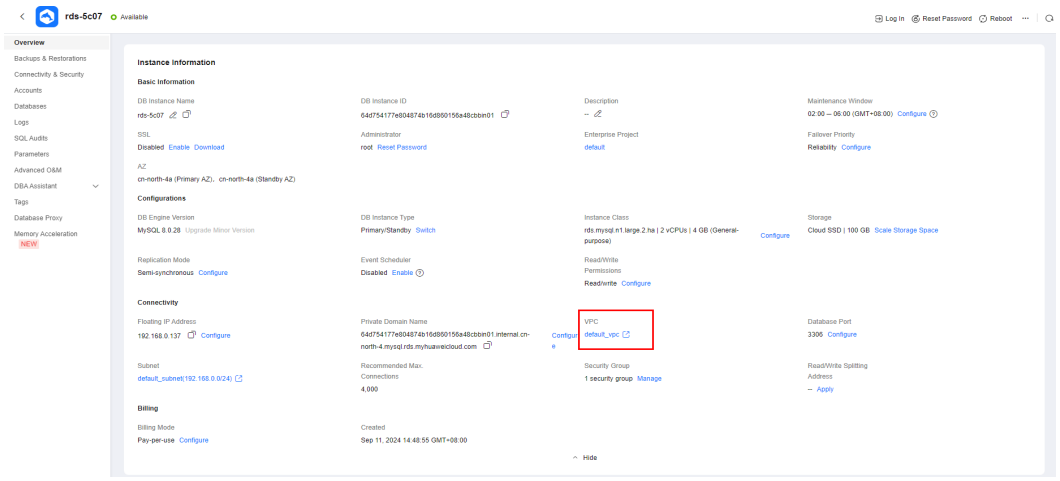
     For details about how to purchase a Linux ECS, see **Purchasing a Custom ECS** in *Elastic Cloud Server User Guide*.
   - If there is only a Windows ECS, you can use MySQL-Front to connect to the DB instance. For details, see **Buying a DB Instance and Connecting to It Using MySQL-Front**.

Figure 1-4 ECS



3. Check whether the ECS and RDS for MySQL instance are in the same region and VPC.

Figure 1-5 Overview

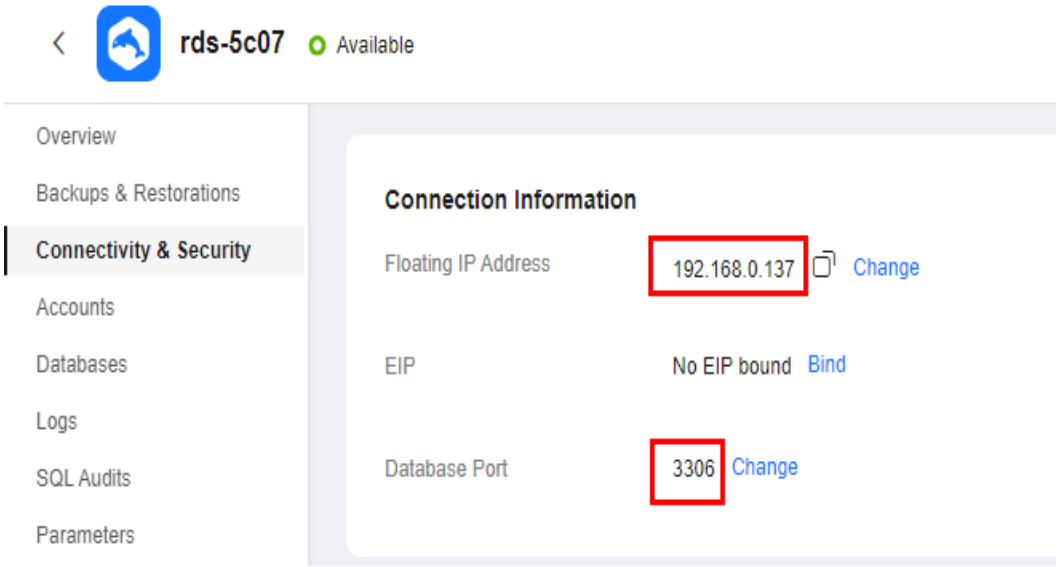


- If they are not in the same region, purchase another ECS. The ECS and DB instance in different regions cannot communicate with each other. To reduce network latency, deploy your DB instance in the region nearest to your workloads.
- If the ECS and DB instance are in different VPCs, change the VPC of the ECS to that of the DB instance. For details, see **Changing a VPC**.

Step 3: Test Connectivity and Install a MySQL Client

1. Log in to the ECS. For details, see **Login Using VNC** in the *Elastic Cloud Server User Guide*.
2. On the **Instances** page of the RDS console, click the DB instance name.
3. Choose **Connectivity & Security** from the navigation pane. In the **Connection Information** area, obtain the floating IP address and database port of the DB instance.

Figure 1-6 Connection information



4. On the ECS, check whether the floating IP address and database port of the DB instance can be connected.

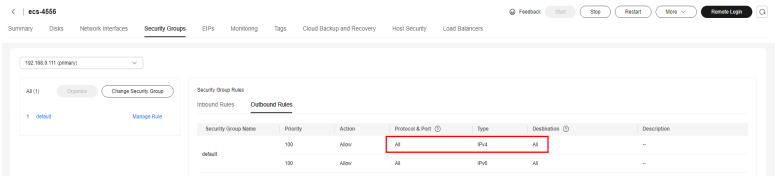
   telnet *192.168.6.144 3306*

   - If yes, network connectivity is normal.
   - If no, check the security group rules.

     - Check the outbound rules of the ECS security group. By default, all outgoing network traffic is allowed.

       If not all outgoing traffic is allowed, add an outbound rule for the floating IP address and port of the DB instance.

       Figure 1-7 ECS security group

     - If in the security group of the DB instance, there is no inbound rule allowing the access from the private IP address and port of the ECS, add an inbound rule for the private IP address and port of the ECS. For details, see **Configuring a Security Group Rule**.

5. Download the MySQL client installation package for Linux to the ECS. The package **mysql-community-client-8.0.28-1.el6.x86_64.rpm** is used as an example.

   wget https://dev.mysql.com/get/mysql-community-client-8.0.28-1.el6.x86_64.rpm

   NOTE

   A MySQL client running a version later than that of the DB instance is recommended.

6. Install the MySQL client.

   rpm -ivh --nodeps mysql-community-client-8.0.28-1.el6.x86_64.rpm

   NOTE

   - If any conflicts occur during the installation, add the **replacefiles** parameter to the command and install the client again.

     rpm -ivh --replacefiles mysql-community-client-8.0.28-1.el6.x86_64.rpm
   - If a message is displayed prompting you to install a dependent package during the installation, add the **nodeps** parameter to the command and install the client again.

     rpm -ivh --nodeps mysql-community-client-8.0.28-1.el6.x86_64.rpm

Step 4: Connect to the DB Instance Using a CLI (Non-SSL Connection)

1. Run the following command on the ECS to connect to the DB instance:

   mysql -h <*host*> **-P** <*port*> **-u** <*userName*> **-p**

   Example:

   mysql -h 192.168.6.144 -P 3306 -u root -p

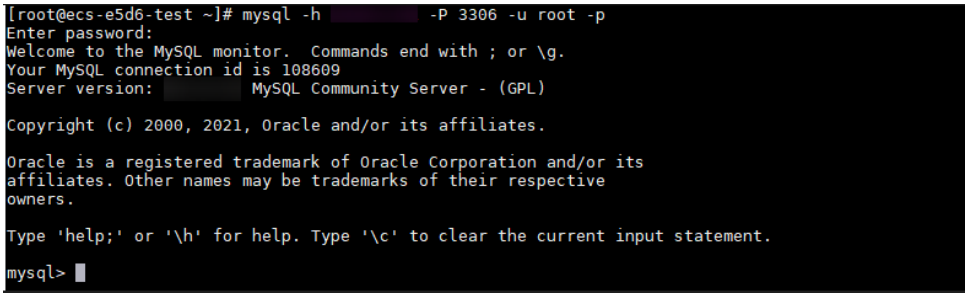
Table 1-1 Parameter description

Parameter	Description
<*host*>	Floating IP address obtained in **3**.
<*port*>	Database port obtained in **3**. The default value is 3306.
<*userName*>	Administrator account **root**.

2. Enter the password of the database account if the following information is displayed:

```
Enter password:
```

Figure 1-8 Connection successful



FAQ

What Should I Do If I Can't Connect to My RDS DB Instance?

Follow-up Operations

After logging in to the DB instance, you can create or migrate databases.

- **Creating a MySQL Database Using the Console**
- **Creating a MySQL Database Using an API**
- **Managing MySQL Databases Using DAS**
- **Migrating Data to RDS for MySQL**

1.2 Buying a DB Instance and Connecting to It Using MySQL-Front

After buying a DB instance, you can connect to it using a Windows ECS with MySQL-Front installed over a private network.

MySQL-Front is a Windows front end for MySQL databases. It allows you to interact with MySQL databases through a GUI. For example, you can connect to a database, run SQL commands, and manage tables and records.

Operation Process

Process	Description
Preparations	Sign up for a HUAWEI ID, enable Huawei Cloud services, make sure you have a valid payment method configured, create IAM users, and grant them specific RDS permissions.
Step 1: Buy an RDS for MySQL DB Instance	Select required basic settings and additional options and buy an RDS for MySQL DB instance.
Step 2: Buy an ECS	If you want to use MySQL-Front to connect to a DB instance, you need to prepare a server, install MySQL-Front on the server, and log in to the instance. Purchase a Windows ECS that is in the same region and VPC as your DB instance. If you have purchased a Linux ECS, you can connect to the DB instance using a MySQL client. For details, see Buying a DB Instance and Connecting to It Using a MySQL Client.
Step 3: Test Connectivity and Install MySQL-Front	Test the network connectivity between the ECS and the floating IP address and port of the DB instance, and install MySQL-Front on the ECS.
Step 4: Connect to the DB Instance Using MySQL-Front	Use MySQL-Front to connect to the DB instance using the floating IP address and port.

Preparations

1. Sign up for a HUAWEI ID and enable Huawei Cloud services.
2. Before purchasing DB instances, ensure that your account balance is sufficient. Top up your account if required.
3. For fine-grained permissions management on Huawei Cloud resources, use Identity and Access Management (IAM) to create a user or user group and grant it specific operation permissions. For details, see Creating a User and Granting Permissions.

Step 1: Buy an RDS for MySQL DB Instance

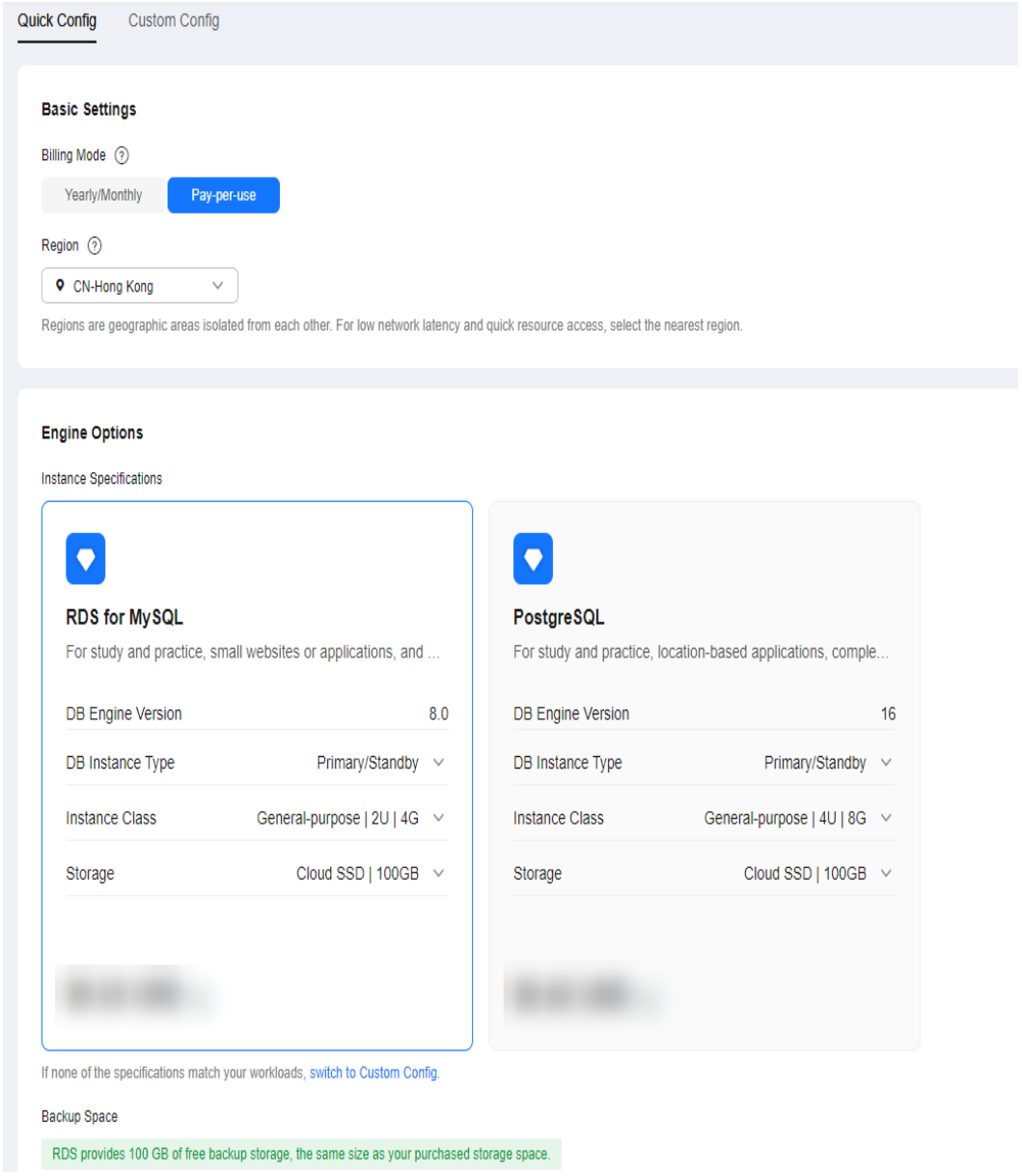
1. Go to the Buy DB Instance page.
2. On the **Quick Config** page, set basic parameters.

   NOTE

   Only mandatory parameters are provided on the **Quick Config** page. If the available parameters do not match your workloads, try Custom Config.

   The following parameter settings are only for reference.

Figure 1-9 Basic Settings

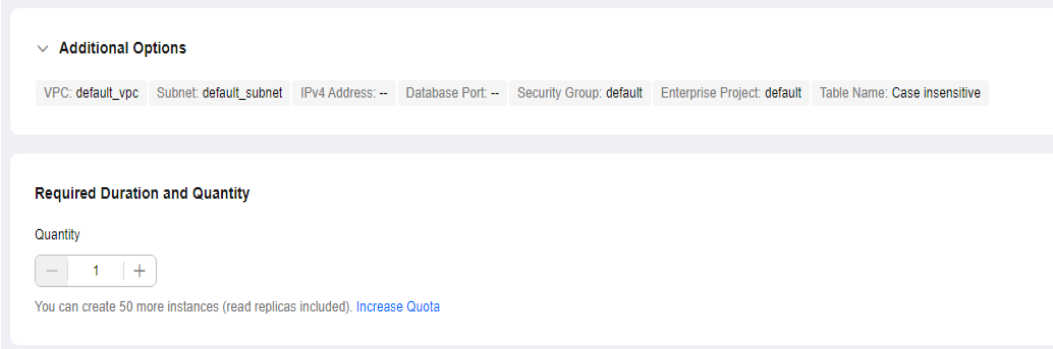


Parameter	Example Value	Description
Billing Mode	Pay-per-use	The billing mode of an instance. • **Yearly/Monthly**: A prepaid billing mode in which you pay for resources before using it. Bills are settled based on the subscription period. The longer the subscription, the bigger the discount. This mode is a good option for long-term, stable services. • **Pay-per-use**: A postpaid billing mode. You pay as you go and just pay for what you use. The DB instance usage is calculated by the second but billed every hour. This mode allows you to adjust resource usage easily. You neither need to prepare for resources in advance, nor end up with excessive or insufficient preset resources.
Region	CN-Hong Kong	The region where your resources are located. **NOTE** Products in different regions cannot communicate with each other through a private network. After a DB instance is created, the region cannot be changed. Therefore, exercise caution when selecting a region.
DB Engine Version	8.0	The database version.
DB Instance Type	Primary/ Standby	The architecture type of an instance. **Primary/Standby**: An HA architecture. In a primary/standby pair, each instance has the same instance class. When a primary instance is being created, a standby instance is provisioned along with it to provide data redundancy. The standby instance is invisible to you after being created.
Instance Class	General-purpose | 2U | 4G	The vCPU and memory of an instance.
Storage	Cloud SSD | 100GB	The storage space of an instance. It contains the system overhead required for inodes, reserved blocks, and database operation.

Parameter	Example Value	Description
Disk Encryption	Disable	Enabling disk encryption improves data security, but slightly affects the read and write performance of the database. If a shared KMS key is used, the corresponding CTS events are **createdatakey** and **decrydatakey**. Only the key owner can receive the events.

3. Complete advanced configurations.

Figure 1-10 Additional Options

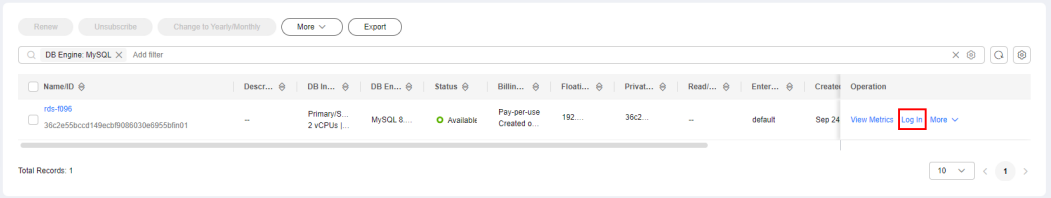


Parameter	Example Value	Description
VPC	vpc-default	The virtual network in which your instance is located. A VPC can isolate networks for different workloads. If no VPC is available, click **Create VPC**. After a VPC is created, click . For details, see **Creating a VPC and Subnet**.
Subnet	subnet-default	A subnet provides dedicated network resources that are logically isolated from other networks for network security.
Security Group	default	It can enhance security by controlling access to RDS for MySQL from other services.
Enterprise Project	default	If your account has been associated with an enterprise project, select the target project from the **Enterprise Project** drop-down list. For more information about enterprise projects, see ***Enterprise Management User Guide***.

Parameter	Example Value	Description
Table Name	Case insensitive	Whether table names are case sensitive. Restoration may fail if the case sensitivity settings of table names on the source and target instances are different. The case sensitivity of table names for created RDS for MySQL 8.0 instances cannot be changed.
Quantity	1	The number of instances to be created in a batch.

4. Click **Buy**.
5. Check the purchased DB instance.

Figure 1-11 Instance successfully purchased



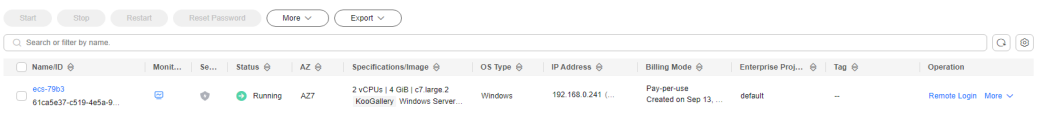
Step 2: Buy an ECS

1. Go to the **Elastic Cloud Server console**.
2. Check whether there is a Windows ECS.
   - If yes, go to **3**.
   - If no, purchase an ECS and select Windows as its OS.

     To download MySQL-Front to the ECS, bind an EIP to the ECS. The ECS must be in the same region, VPC, and security group as the RDS for MySQL DB instance for mutual communications.

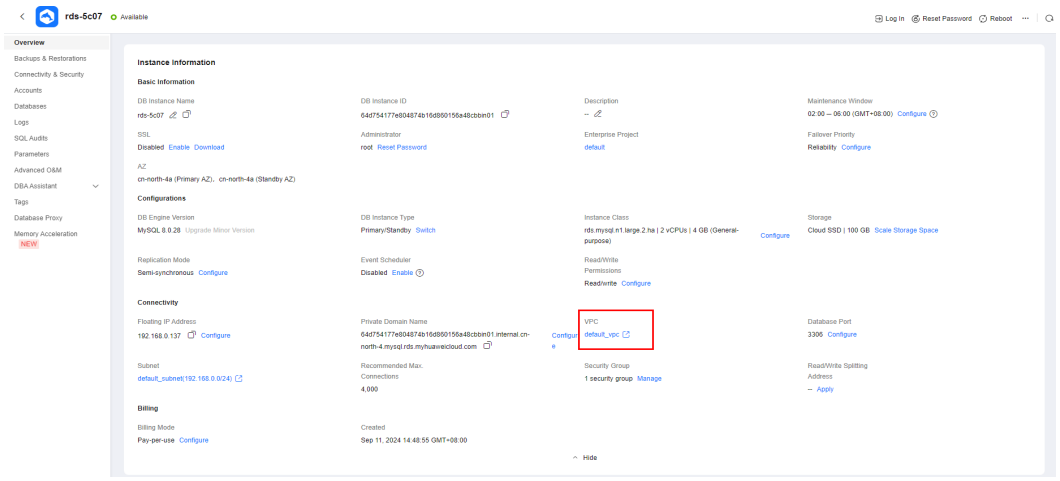
     For details about how to purchase a Windows ECS, see **Purchasing a Custom ECS** in *Elastic Cloud Server User Guide*.
   - If there is only a Linux ECS, you can use a MySQL client to connect to the DB instance. For details, see **Buying a DB Instance and Connecting to It Using a MySQL Client**.

Figure 1-12 ECS



3. Check whether the ECS and RDS for MySQL instance are in the same region and VPC.

Figure 1-13 Overview

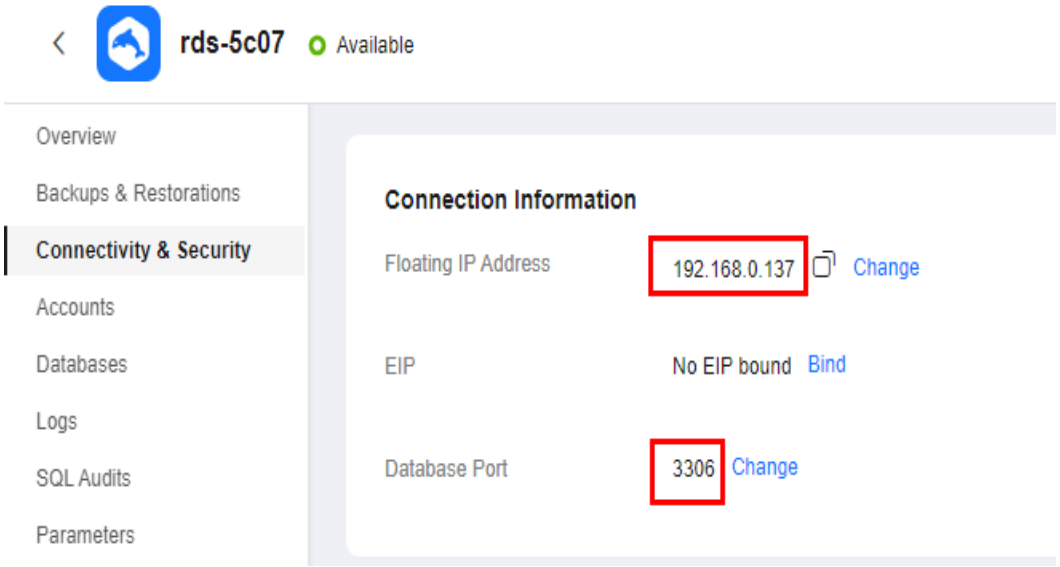


- If they are not in the same region, purchase another ECS. The ECS and DB instance in different regions cannot communicate with each other. To reduce network latency, deploy your DB instance in the region nearest to your workloads.
- If the ECS and DB instance are in different VPCs, change the VPC of the ECS to that of the DB instance. For details, see **Changing a VPC**.

Step 3: Test Connectivity and Install MySQL-Front

1. Log in to the ECS. For details, see **Login Using VNC** in the *Elastic Cloud Server User Guide*.
2. On the **Instances** page of the RDS console, click the DB instance name.
3. Choose **Connectivity & Security** from the navigation pane. In the **Connection Information** area, obtain the floating IP address and database port of the DB instance.

Figure 1-14 Connection information



4. Open the cmd window on the ECS and check whether the floating IP address and database port of the DB instance can be connected.

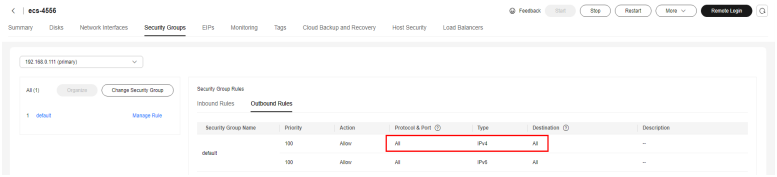
   telnet *192.168.6.144 3306*

   - If yes, network connectivity is normal.
   - If no, check the security group rules.

     - Check the outbound rules of the ECS security group. By default, all outgoing network traffic is allowed.

       If not, add an outbound rule for the floating IP address and port of the DB instance.

       Figure 1-15 ECS security group

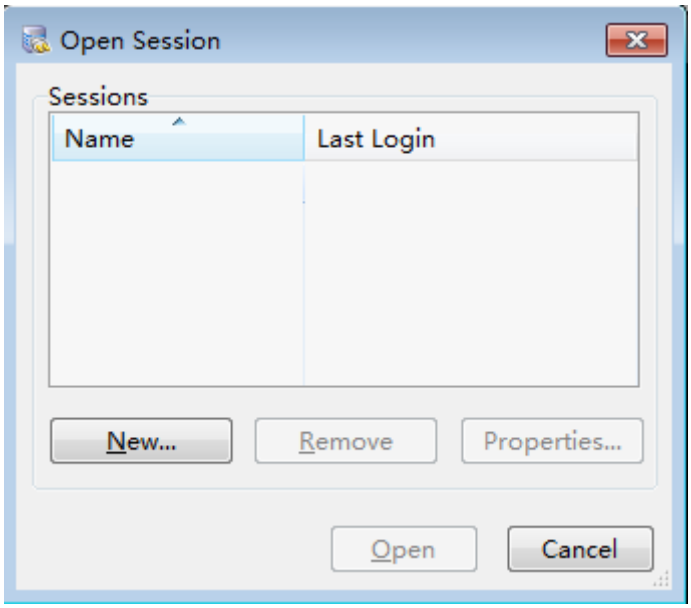
     - If in the security group of the DB instance, there is no inbound rule allowing the access from the private IP address and port of the ECS, add an inbound rule for the private IP address and port of the ECS. For details, see **Configuring a Security Group Rule**.

5. Open a browser, and download and install the MySQL-Front tool on the ECS (version 5.4 is used as an example).

Step 4: Connect to the DB Instance Using MySQL-Front

1. Start MySQL-Front.
2. In the displayed dialog box, click **New**.

   Figure 1-16 Connection management

3. Enter the information of the DB instance to be connected and click **Ok**.

Figure 1-17 Adding an account

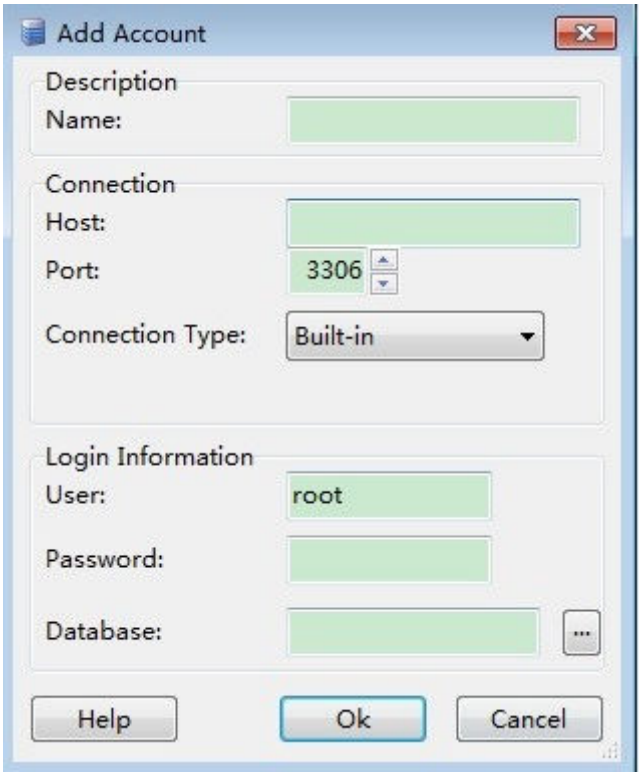
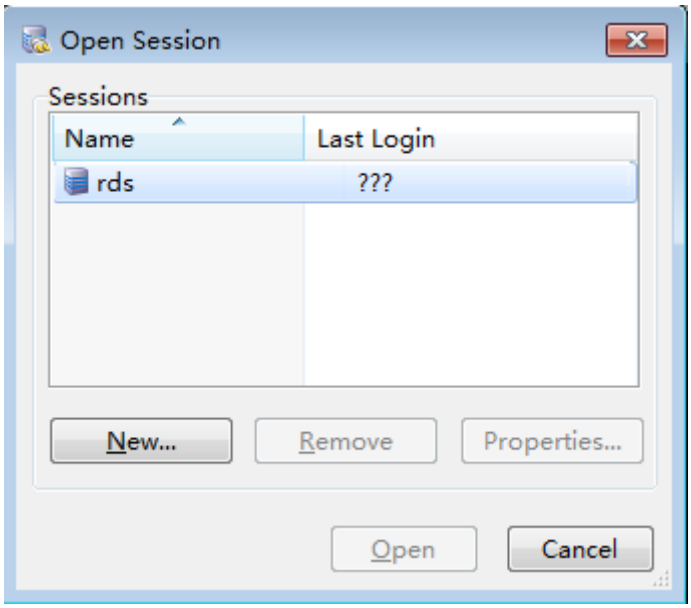


Table 1-2 Parameter description

Parameter	Description
Name	Name of the database connection task. If you do not specify this parameter, it will be the same as that configured for **Host** by default.
Host	Floating IP address obtained in **3**.
Port	Database port obtained in **3**. The default value is 3306.
User	Name of the user who will access the DB instance. The default user is **root**.
Password	Password of the account for accessing the DB instance.

4. In the displayed window, select the connection that you have created in **3** and click **Open**. If the connection information is correct, the DB instance will be connected.

Figure 1-18 Opening a session



FAQs

What Should I Do If I Can't Connect to My RDS DB Instance?

Follow-up Operations

After logging in to the DB instance, you can create or migrate your databases.

- **Creating a Database Using the Console**
- **Creating a Database Using an API**
- **Managing Databases Using DAS**
- **Migrating Data to RDS for MySQL**

1.3 Getting Started with RDS for MySQL Common Practices

After purchasing and connecting to a DB instance, you can view common practices to better use RDS for MySQL.

Table 1-3 Common practices

Practice		Description
Suggestions on using RDS for MySQL	**Instance Usage Suggestions**	This practice provides suggestions on using RDS for MySQL in terms of DB instances, database connection, reliability and availability, backup and restoration, SQL audit, routine O&M, and security.

Practice		Description
	Database Usage Suggestions	This practice provides suggestions on database naming, database design, field design, index design, and SQL statement development.
Website setup	**Using RDS for MySQL to Set Up WordPress**	This practice describes how to set up WordPress in a LAMP environment using Huawei Cloud Virtual Private Cloud (VPC), Elastic Cloud Server (ECS), and RDS for MySQL.
	Using RDS for MySQL to Set Up Discuz!	This practice describes how to set up Discuz! in a LAMP environment using Huawei Cloud VPC, ECS, and RDS for MySQL.
Data migration	**Migrating Data to RDS for MySQL Using mysqldump**	This practice describes how to use mysqldump to copy data from the source to an RDS for MySQL DB instance.
	From RDS for MySQL to RDS for MySQL	This practice describes how to use Data Replication Service (DRS) to migrate table, database, or instance data of the source to an RDS for MySQL DB instance.
	Migrating Data to RDS for MySQL Using the Export and Import Functions of DAS	This practice describes how to use Data Admin Service (DAS) to export data from the source and then import the data to an RDS for MySQL DB instance.
	From RDS for MySQL to RDS for MySQL	This practice describes how to use DRS to synchronize data from the source to an RDS for MySQL DB instance.
	• **Configuring Remote Single-Active DR for an RDS for MySQL Instance Using DRS** • **From RDS for MySQL to RDS for MySQL (Dual-Active DR)**	This practice describes how to use DRS to synchronize data from the source to a DR RDS for MySQL instance.
	From DDM to RDS for MySQL	This practice describes how to use DRS to synchronize data from a DDM instance to an RDS for MySQL DB instance.

Practice		Description
	From GaussDB Distributed to RDS for MySQL	This practice describes how to use DRS to synchronize data from a GaussDB distributed instance to an RDS for MySQL DB instance.
	From GaussDB Primary/Standby to RDS for MySQL	This practice describes how to use DRS to synchronize data from a GaussDB primary/standby instance to an RDS for MySQL DB instance.
	From GaussDB(for MySQL) to RDS for MySQL	This practice describes how to use DRS to synchronize data from a GaussDB(for MySQL) DB instance to an RDS for MySQL DB instance.
	Migrating Data from Self-Managed MySQL Databases to RDS for MySQL	This practice describes how to use DRS to migrate data from a self-managed MySQL database to an RDS for MySQL DB instance.
	From Self-Managed MySQL to RDS for MySQL	This practice describes how to use DRS to synchronize data from a self-managed MySQL database to an RDS for MySQL DB instance.
	• From Self-Managed MySQL to RDS for MySQL (Single-Active DR) • From Self-Managed MySQL to RDS for MySQL (Dual-Active DR)	This practice describes how to use DRS to synchronize data from a self-managed MySQL database to a DR RDS for MySQL instance.
	From Oracle to RDS for MySQL	This practice describes how to use DRS to synchronize data from a self-managed Oracle database to an RDS for MySQL DB instance.
	Migrating MySQL Databases from Other Clouds to RDS for MySQL	This practice describes how to use DRS to migrate MySQL databases from other clouds to RDS for MySQL.
	From MySQL on Other Clouds to RDS for MySQL	This practice describes how to use DRS to synchronize MySQL databases from other clouds to RDS for MySQL.

Practice		Description
	• From MySQL on Other Clouds to RDS for MySQL (Single-Active DR) • From MySQL on Other Clouds to RDS for MySQL (Dual-Active DR)	This practice describes how to use DRS to synchronize MySQL databases from other clouds to DR RDS for MySQL instances.
Data backup	Intra-region automated backup	This practice describes how RDS for MySQL automatically creates backups for a DB instance during a backup window and saves the backups based on the configured retention period.
	Intra-region manual backup	This practice describes how to create manual backups for a DB instance. These backups can be used to restore data for improved reliability.
	Cross-region automated backup	This practice describes how to store backups of a DB instance in another region for disaster recovery. If the DB instance fails, the backups in another region can be used to restore the data to a new DB instance.
Data restoration	Restoring from Full Backups to RDS for MySQL Instances	This practice describes how to use an automated or manual backup to restore a DB instance to how it was when the backup was created. The restoration is at the instance level.
	Restoring a DB Instance to a Point in Time	This practice describes how to use an automated backup to restore instance data to a specified point in time.
	Restoring Databases or Tables to a Point in Time	This practice describes how to use an automated backup to restore databases or tables to a specified point in time.

2 Getting Started with RDS for MariaDB

2.1 Step 1: Set Up for RDS

You can buy and connect to DB instances on the RDS console.

Registering a HUAWEI ID

If you already have a HUAWEI ID, skip this part. If you do not have a HUAWEI ID yet, perform the following steps to create one:

Step 1 Open the **Huawei Cloud website**.

Step 2 Click **Register** and complete the registration as instructed.

After the registration is successful, the system redirects you to your personal information page.

----End

Topping Up Your Account

- For details about RDS for MariaDB prices, see **Price Calculator**.
- Before purchasing an RDS for MariaDB instance, ensure that your account balance is sufficient. For details about how to top up an account, see **Topping Up an Account**.

Creating an IAM User and Granting Permissions

You can create an IAM user or user group on the Identity and Access Management (IAM) console and grant it specific operation permissions for fine-grained permissions management.

1. **Create a user group and assign permissions** to it.

   Create a user group on the IAM console, and attach the **RDS ReadOnlyAccess** policy to the group.

NOTE

To use some interconnected services, you also need to configure permissions of such services.

For example, to connect to your DB instance through the console, configure the **DAS FullAccess** permission of Data Admin Service (DAS) besides **RDS ReadOnlyAccess**.

2. **Create an IAM user and add it to the user group**.

   Create a user on the IAM console and add the user to the group created in **1**.

3. **Log in** and verify permissions.

   Log in to the RDS console by using the created user, and verify that the user only has read permissions for RDS.

   - Choose **Service List** > **Relational Database Service** and click **Buy DB Instance**. If a message appears indicating that you have insufficient permissions to perform the operation, the **RDS ReadOnlyAccess** policy has already been applied.
   - Choose any other service in **Service List**. If a message appears indicating that you have insufficient permissions to access the service, the **RDS ReadOnlyAccess** policy has already taken effect.

2.2 Step 2: Buy a DB Instance

Scenarios

This section describes how to buy a DB instance on the RDS console.

RDS for MariaDB only supports the pay-per-use billing mode. RDS allows you to tailor your compute resources and storage space to your business needs.

Prerequisites

Your account balance is greater than or equal to $0 USD.

Procedure

Step 1 Go to the **Buy DB Instance** page.

Step 2 On that page, select a billing mode, configure information about your DB instance, and click **Next**.

- Basic information

Figure 2-1 Basic information

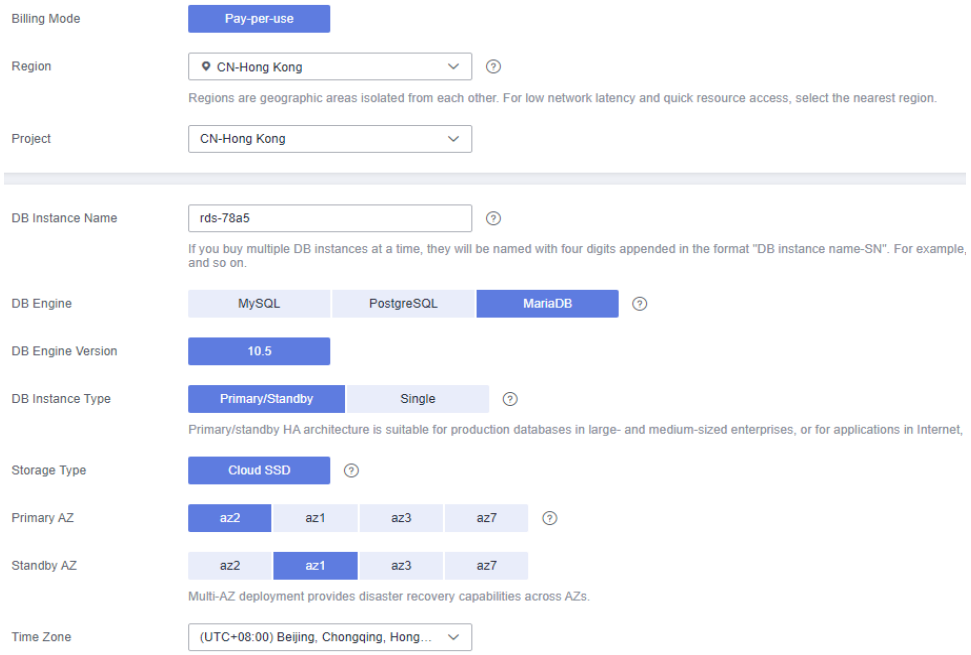


Table 2-1 Basic information

Parameter	Description
Billing Mode	Select **Pay-per-use**.
Region	Region where your resources are located. **NOTE** Products in different regions cannot communicate with each other through a private network. After a DB instance is created, the region cannot be changed. Therefore, exercise caution when selecting a region.
Project	The project corresponds to the region. Different regions correspond to different projects.
DB Instance Name	Must start with a letter and consist of 4 to 64 characters. Only letters (case-sensitive), digits, hyphens (-), and underscores (_) are allowed. – If you intend to buy multiple DB instances at a time, the allowed length for each instance name will change. – If you buy multiple DB instances at a time, they will be named *instance-0001*, *instance-0002*, and so on. (*instance* indicates the DB instance name you specify.)
DB Engine	MariaDB
DB Engine Version	For details, see **DB Engines and Versions**. The DB engine version differs in different regions.

Parameter	Description
DB Instance Type	– **Primary/Standby**: uses an HA architecture with a primary DB instance and a synchronous standby DB instance. It is suitable for production databases of large- and medium-sized enterprises in Internet, Internet of Things (IoT), retail e-commerce sales, logistics, gaming, and other sectors. The standby DB instance improves instance reliability and is invisible to you after being created. – **Single**: uses a single-node architecture, which is more cost-effective than primary/standby DB instances. It is only recommended for development and testing of microsites, and small and medium enterprises, or for learning about RDS.
AZ	An AZ is a physical region where resources use independent power supply and networks. AZs are physically isolated but interconnected through an internal network. Some regions support both single-AZ and multi-AZ deployment and some only support single-AZ deployment. To achieve high reliability, RDS will automatically deploy your primary and standby instances in different physical servers even if you deploy them in the same AZ. You can deploy primary and standby instances in a single AZ or across AZs to achieve failover and high availability.
Storage Type	Determines the DB instance read/write speed. The higher the maximum throughput is, the higher the DB instance read/write speed can be. – **Cloud SSD**: cloud drives used to decouple storage from compute. The maximum throughput is 350 MB/s. – **Extreme SSD**: uses 25GE network and RDMA technologies to provide you with up to 1,000 MB/s throughput per disk and sub-millisecond latency.
Time Zone	You need to select a time zone for your instance based on the region hosting your instance. You can select a time zone during instance creation and change it later as needed.

- Specifications and storage

Figure 2-2 Specifications and storage

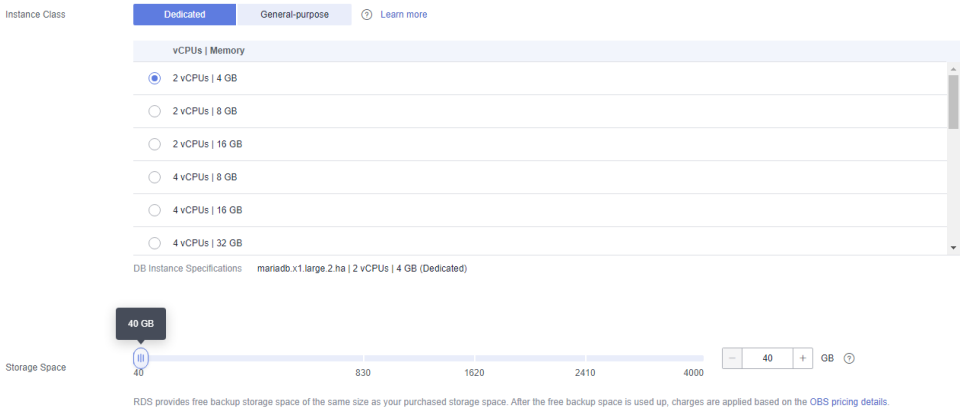


Table 2-2 Specifications and storage

Parameter	Description
Instance Class	Refers to the vCPU and memory of a DB instance. Different instance classes support different numbers of database connections and maximum IOPS. After a DB instance is created, you can change its vCPU and memory. For details, see **Changing a DB Instance Class**.
Storage Space (GB)	Contains the system overhead required for inodes, reserved blocks, and database operation. Storage space can range in size from 40 GB to 4,000 GB and can be scaled up only by a multiple of 10 GB. After a DB instance is created, you can scale up its storage space. For details, see **Scaling up Storage Space**.

- Network and database configuration

Figure 2-3 Network and database configuration

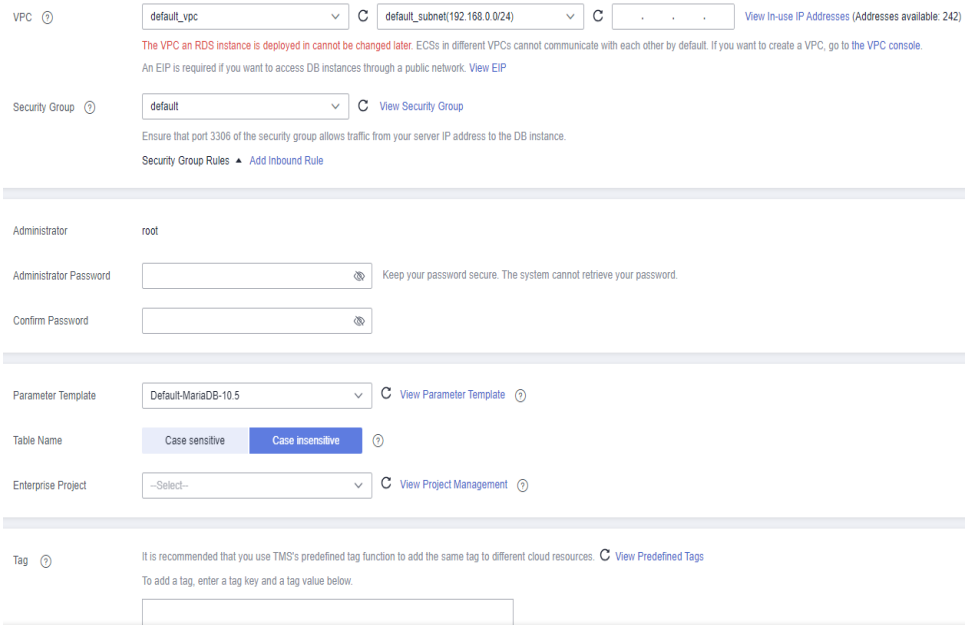


Table 2-3 Network

Parameter	Description
VPC	A virtual network in which your RDS DB instances are located. A VPC can isolate networks for different workloads. You can select an existing VPC or create a VPC. For details about how to create a VPC, see "Creating a VPC" in *Virtual Private Cloud User Guide*. If no VPC is available, RDS allocates a VPC to you by default. **NOTICE** After a DB instance is created, the VPC cannot be changed.
Subnet	Improves network security by providing dedicated network resources that are logically isolated from other networks. Subnets take effect only within an AZ. The Dynamic Host Configuration Protocol (DHCP) function is enabled by default for subnets in which you plan to create RDS DB instances and cannot be disabled. A floating IP address is automatically assigned when you create a DB instance. You can also enter an unused IPv4 floating IP address in the subnet CIDR block.
Security Group	Enhances security by controlling access to RDS from other services. A network **access control list (ACL)** can help control inbound and outbound traffic of subnets in your VPC. Ensure that the security group you select allows the client to access the DB instance. If no security group is available or has been created, RDS allocates a security group to you by default.

Table 2-4 Database configuration

Parameter	Description
Administrator	The default login name for the database is **root**.
Administrator Password	Must consist of 8 to 32 characters and contain the following character types: uppercase letters, lowercase letters, digits, and special characters (~!@#$%^*-_=+?,()&). Enter a strong password and periodically change it for security reasons. If the password you provide is regarded as a weak password by the system, you will be prompted to enter a stronger password. Keep this password secure. The system cannot retrieve it. After a DB instance is created, you can reset this password. For details, see **Resetting the Administrator Password**.
Confirm Password	Must be the same as **Administrator Password**.
Parameter Template	Contains engine configuration values that can be applied to one or more DB instances. If you intend to create a primary/standby DB pair, they use the same parameter template. You can modify the instance parameters as required after the DB instance is created. For details, see **Modifying Parameters in a Parameter Template**.
Table Name	Specifies whether table names are case sensitive. **NOTE** The case sensitivity of table names for created instances cannot be changed.
Enterprise Project	If your account has been associated with an enterprise project, select the target project from the **Enterprise Project** drop-down list. For more information about enterprise projects, see ***Enterprise Management User Guide***.

- Tags

Table 2-5 Tags

Parameter	Description
Tag	Tags an RDS DB instance. This parameter is optional. Adding tags to RDS DB instances helps you better identify and manage the DB instances. A maximum of 20 tags can be added for each DB instance. If your organization has configured tag policies for RDS, add tags to DB instances based on the policies. If a tag does not comply with the policies, DB instance creation may fail. Contact your organization administrator to learn more about tag policies. After a DB instance is created, you can view its tag details on the **Tags** page. For details, see **Managing Tags**.

- Purchase period

Table 2-6 Purchase period

Parameter	Description
Quantity	RDS supports DB instance creation in batches. If you choose to create primary/standby DB instances and set **Quantity** to **1**, a primary DB instance and a standby DB instance will be created synchronously.

NOTE

- If you have any questions about the price, move the cursor to in the **Price** area at the bottom of the page and click **Pricing details**.
- The performance of your DB instance depends on its configurations. Hardware configuration items include the instance specifications, storage type, and storage space.

Step 3 Confirm the specifications.

- If you do not need to modify your settings, click **Submit**.
- If you need to modify your settings, click **Previous**.

Step 4 To view and manage your DB instance, go to the **Instances** page.

- When your DB instance is being created, the status is **Creating**. The status changes to **Available** after the instance is created. To view the detailed progress and result of the creation, go to the **Task Center** page.
- The automated backup policy is enabled by default. You can change it after the DB instance is created. An automated full backup is immediately triggered once your DB instance is created.
- After a DB instance is created, you can enter a description for it.
- The default database port is **3306**. You can change it after a DB instance is created.

NOTE

You are advised to change the default database port in a timely manner. For details, see **Changing a Database Port**.

----End

2.3 Step 3: Connect to a DB Instance

2.3.1 Overview

An RDS DB instance can be connected through a private network, Data Admin Service (DAS), or a public network.

Table 2-7 RDS connection methods

Connect Through	IP Address	Scenarios	Description
DAS	No IP address required	DAS enables you to manage databases on a web-based console and provides you with database development, O&M, and intelligent diagnosis to make it easy to use and maintain your databases. The permissions required for connecting to DB instances through DAS are enabled by default.	• Easy to use, secure, advanced, and intelligent • Recommended
Private network	Floating IP	RDS provides a floating IP address by default. If your applications are deployed on an ECS that is in the same region and VPC as your DB instance, you are advised to use a floating IP address to connect to the DB instance through the ECS.	• Secure and excellent performance • Recommended

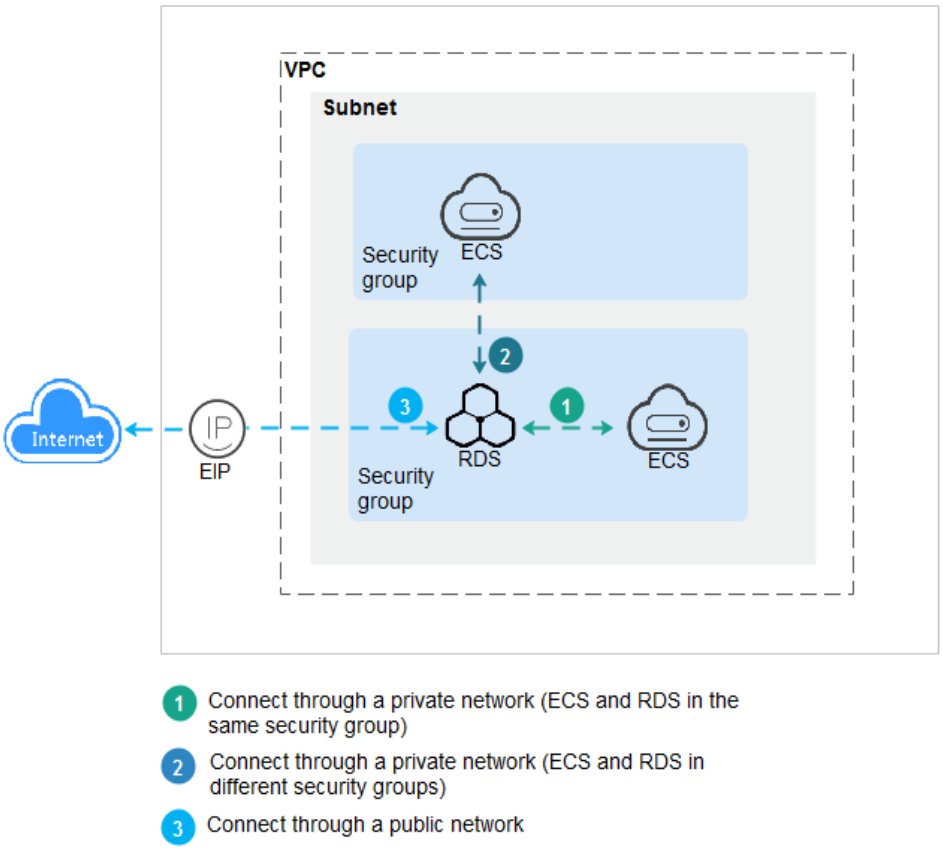
Connect Through	IP Address	Scenarios	Description
Public network	EIP	If you cannot access a DB instance through a floating IP address, bind an EIP to the DB instance and connect the DB instance through the EIP.	• A relatively lower level of security compared to other connection methods • To achieve a higher transmission rate and security level, you are advised to migrate your applications to an ECS that is in the same VPC as your RDS DB instance and use a floating IP address to access the DB instance. • You need to purchase an EIP. For details, see **EIP billing details**.

NOTE

- VPC: Virtual Private Cloud
- ECS: Elastic Cloud Server
- If the ECS is in the same VPC as your RDS DB instance, you do not need to apply for an EIP.

Figure 2-4 illustrates the connection over a private network or a public network.

Figure 2-4 DB instance connection



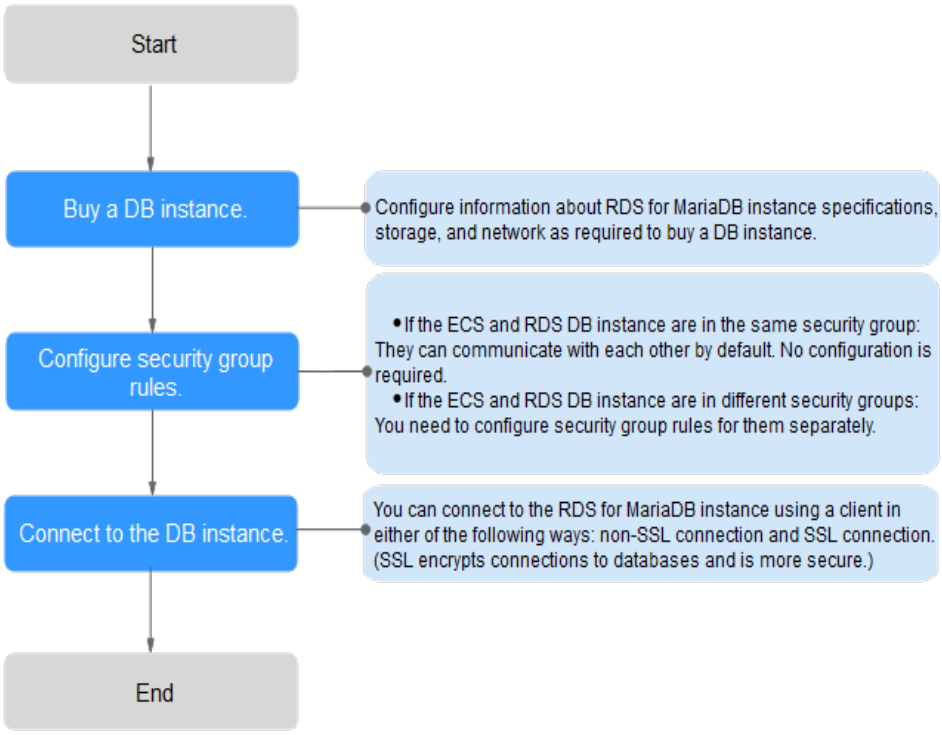
2.3.2 Connecting to a DB Instance Through a Private Network

2.3.2.1 Overview

Process

Figure 2-5 illustrates the process of connecting to an RDS for MariaDB instance through a private network.

Figure 2-5 Connecting to a DB instance through a private network



2.3.2.2 Configuring Security Group Rules

Before you can connect to your DB instance, you need to create security group rules to enable specific IP addresses and ports to access your RDS DB instance. This section describes how to configure an inbound rule for a DB instance.

Context

A security group is a collection of access control rules for ECSs and RDS DB instances that have the same security protection requirements and are mutually trusted in a VPC.

Scenarios

First check whether the ECS and RDS DB instance are in the same security group.

- If they are in the same security group, they can communicate with each other by default. No security group rule needs to be configured. Go to **Connecting to a DB Instance Using a MariaDB Client**.
- If they are in different security groups, configure security group rules for them, separately.
  - RDS DB instance: Configure an **inbound rule** for the security group with which the RDS DB instance is associated.
  - ECS: The default security group rule allows all outgoing data packets. In this case, you do not need to configure a security group rule for the ECS. If not all outbound traffic is allowed in the security group, you need to configure an **outbound rule** for the ECS.

For details about the requirements of security group rules, see **Adding a Security Group Rule** in *Virtual Private Cloud User Guide*.

Constraints

The default security group rule allows all outgoing data packets. ECSs and RDS DB instances can access each other if they are in the same security group. After a security group is created, you can configure security group rules to control access from and to the DB instances in the security group.

- By default, you can create a maximum of 100 security groups in your cloud account.
- By default, you can add up to 50 security group rules to a security group.
- One RDS instance can be associated only with one security group, but one security group can be associated with multiple RDS instances.
- Too many security group rules will increase the first packet latency. You are advised to create no more than 50 rules for a security group.
- To enable access to an RDS DB instance from resources outside the security group, you need to configure an **inbound rule** for the security group associated with the RDS DB instance.

NOTE

To ensure the security of your data and DB instances, you are advised to use the principle of least privilege for database access. Change the database port (default value: **3306**), and set the IP address to the remote server's address or any IP address in the remote server's smallest subnet to control the access from the remote server.

The default value of **Source** is **0.0.0.0/0**, indicating that RDS DB instances in the security group can be accessed from any IP address.

Procedure

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

Step 2 Click in the upper left corner and select a region and a project.

Step 3 Click in the upper left corner of the page and choose **Databases** > **Relational Database Service**.

Step 4 On the **Instances** page, click the DB instance name.

Step 5 In the navigation pane on the left, choose **Connectivity & Security**. In the **Security Group Rules** area, view security group rules.

Step 6 Click **Add Inbound Rule** or **Allow All IP** to configure security group rules.

To add more inbound rules, click .

NOTE

Allow All IP allows all IP addresses to access RDS DB instances in the security group, which poses high security risks. Exercise caution when performing this operation.

Figure 2-6 Adding an inbound rule

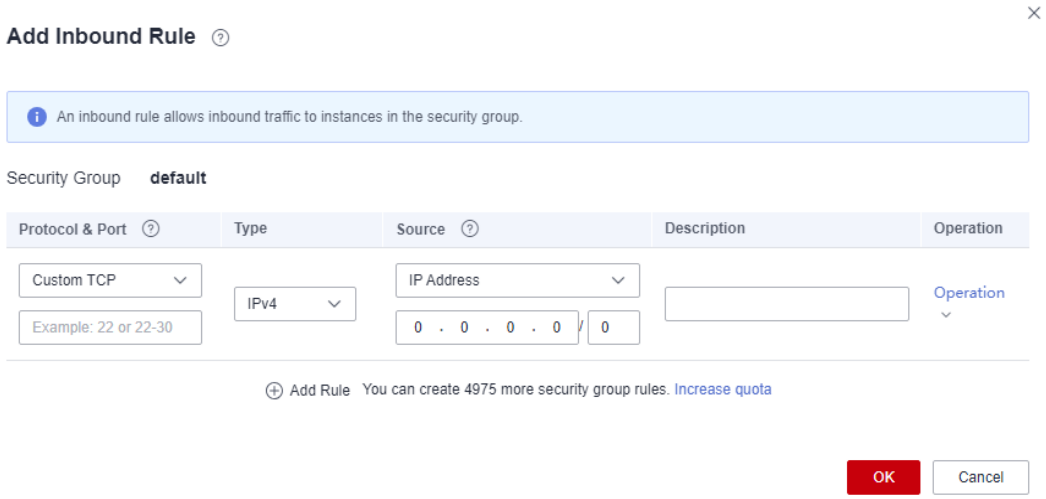


Table 2-8 Inbound rule parameter description

Parameter	Description	Example Value
Protocol & Port	**Protocol**: network protocol. Available options: **All ports**, **Custom TCP**, **Custom UDP**, **ICMP**, or **GRE**.	Custom TCP
	Port: the port over which the traffic can reach your DB instance. RDS for MariaDB instances can use database ports 1024 to 65535, excluding 12017 and 33071, which are reserved for RDS system use.	3306
Type	Supported source IP address type. Its value can be: • IPv4 • IPv6	IPv4

Parameter	Description	Example Value
Source	The source in an inbound rule is used to match the IP address or address range of an external request. The source can be: • Single IP address: 192.168.10.10/32 (IPv4 address) • IP address segment: 192.168.1.0/24 (IPv4 address segment) • All IP addresses: 0.0.0.0/0 (any IPv4 address) • Security group: sg-abc • IP address group: ipGroup-test	0.0.0.0/0
Description	Supplementary information about the security group rule. This parameter is optional. The description can contain a maximum of 255 characters and cannot contain angle brackets (<) or (>).	N/A

Step 7 Click **OK**.

----End

2.3.2.3 Connecting to a DB Instance Using a MariaDB Client

You can connect to a DB instance through a Secure Sockets Layer (SSL) connection or a non-SSL connection. SSL encrypts connections to your DB instance, making in-transit data more secure.

Prerequisites

1. You have logged in to an ECS.
   - For details on how to create and log in to an ECS, see **Purchasing an ECS** and **Logging In to an ECS**.
   - To connect to a DB instance through an ECS, you must ensure that:
     - The ECS and DB instance are in the same VPC.
     - The ECS is allowed by the security group to access the DB instance.
       - If the security group associated with the DB instance is the default security group, you do not need to configure security group rules.
       - If the security group associated with the DB instance is not the default security group, check whether the security group rules

allow the ECS to connect to the DB instance. For details, see **Configuring Security Group Rules**.

If the rules allow the access from the ECS, you can connect to the DB instance through the ECS.

If the rules do not allow the access from the ECS, you need to add a security group rule allowing the ECS to access the DB instance.

2. You have installed a database client to connect to DB instances.

   You can use a database client to connect to the target DB instance in Linux or Windows.

   - In Linux, install a **MariaDB client** on a device that can access RDS. It is recommended that you download a MariaDB client running a version later than that of the DB instance.
   - In Windows, you can use any common database client to connect to the target DB instance in a similar way.

Connecting to a DB Instance Using Commands (SSL Connection)

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

Step 2 Click in the upper left corner and select a region and a project.

Step 3 Click in the upper left corner of the page and choose **Databases** > **Relational Database Service**.

Step 4 On the **Instances** page, click the instance name to go to the **Basic Information** page.

Step 5 In the **DB Information** area, check whether SSL is enabled.

- If yes, go to **Step 6**.
- If no, click . In the displayed dialog box, click **OK**. Then, go to **6**.

Step 6 Click next to the **SSL** field to download **Certificate Download.zip**, and extract the root certificate **ca.pem** and bundle **ca-bundle.pem** from the package.

Step 7 Import the root certificate **ca.pem** to the Linux or Windows. For details, see **How Can I Import the Root Certificate to a Windows or Linux OS?**

Step 8 Connect to the RDS for MariaDB instance. In Linux, for example, run the following command:

mysql -h <*host*> **-P** <*port*> **-u** <*userName*> **-p --ssl-ca=**<*caName*>

Example:

mysql -h 172.16.0.31 -P 3306-u root -p --ssl-ca=ca.pem

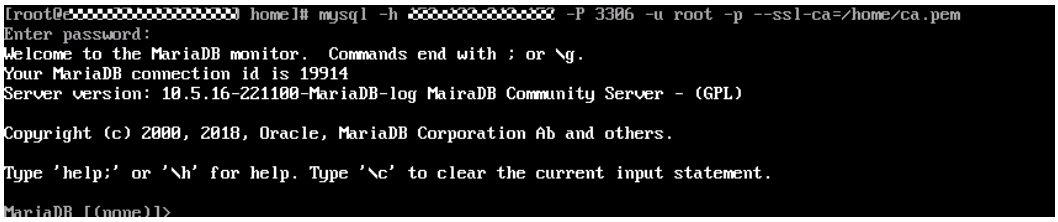
Table 2-9 Parameter description

Parameter	Description
<*host*>	Floating IP address. To obtain this parameter value, go to the **Basic Information** page of the DB instance. You can find the floating IP address in the **Connection Information** area.
<*port*>	Database port. By default, the value is **3306**. To obtain this parameter value, go to the **Basic Information** page of the DB instance. You can find the database port in the **Connection Information** area.
<*userName*>	Database account used for logging in to the DB instance. The default value is **root**.
<*caName*>	Name of the CA certificate. The certificate should be stored in the directory where the command is executed.

Step 9 Enter the password of the database account if the following information is displayed:

```
Enter password:
```

Figure 2-7 Connection example



----End

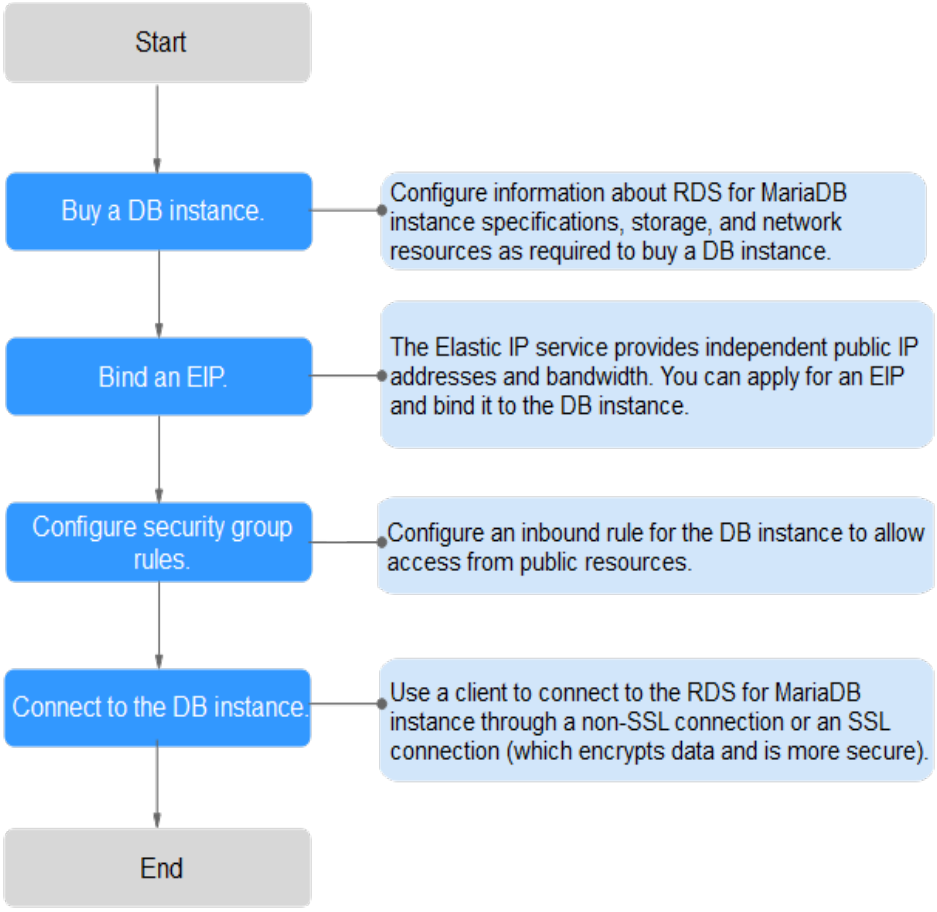
2.3.3 Connecting to a DB Instance Through a Public Network

2.3.3.1 Overview

Process

[Figure 2-8](#) illustrates the process of connecting to an RDS for MariaDB instance through a public network.

Figure 2-8 Connecting to a DB instance through a public network



2.3.3.2 Binding an EIP

Scenarios

You can bind an EIP to a DB instance for public accessibility and can unbind the EIP from the DB instance as required.

Precautions

- To enable this function, contact customer service.
- You need to configure security groups and enable specific IP addresses and ports to access the target DB instance. Before accessing the DB instance, add an individual IP address or an IP address range that will access the DB instance to the inbound rule. For details, see **Configuring Security Group Rules**.
- Traffic generated by the public network is charged. You can unbind the EIP from the DB instance when the EIP is no longer used.

Binding an EIP

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

Step 2 Click in the upper left corner and select a region and a project.

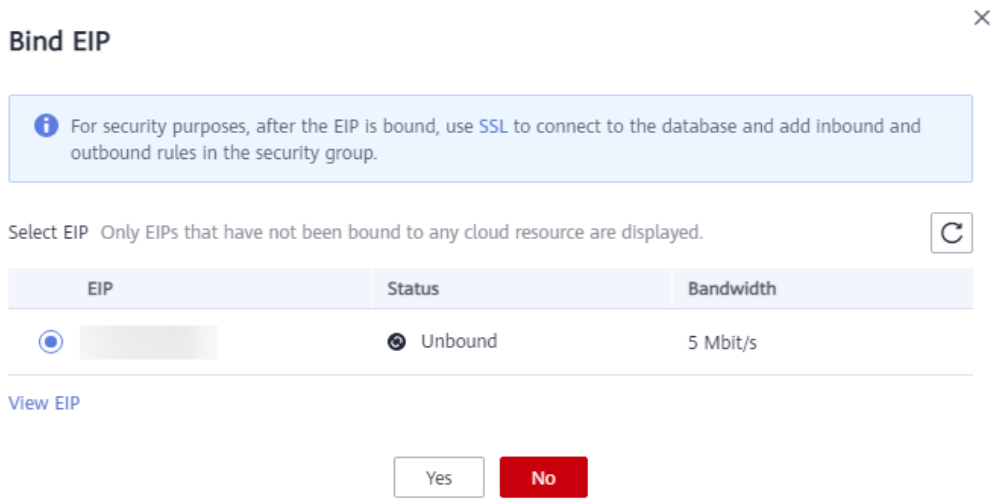
Step 3 Click in the upper left corner of the page and choose **Databases** > **Relational Database Service**.

Step 4 On the **Instances** page, click the target DB instance.

Step 5 In the navigation pane on the left, choose **Connectivity & Security**. In the **Connection Information** area, click **Bind** next to the **EIP** field.

Step 6 In the displayed dialog box, select an EIP and click **Yes**.

Figure 2-9 Selecting an EIP



Step 7 On the **Connectivity & Security** page, view the EIP that has been bound to the DB instance.

You can also view the progress and result of binding an EIP to a DB instance on the **Task Center** page.

----End

2.3.3.3 Configuring Security Group Rules

For security, you need to create security group rules to allow specific IP addresses and ports to access your RDS DB instance. When you attempt to connect to an RDS DB instance through an EIP, configure an **inbound rule** for the security group associated with the DB instance.

Context

A security group is a collection of access control rules for ECSs and RDS DB instances that have the same security protection requirements and are mutually trusted within a VPC.

Constraints

The default security group rule allows all outgoing data packets. ECSs and RDS DB instances can access each other if they are in the same security group. After a security group is created, you can configure security group rules to control access from and to the DB instances in the security group.

- By default, you can create a maximum of 100 security groups in your cloud account.
- By default, you can add up to 50 security group rules to a security group.
- One RDS instance can be associated only with one security group, but one security group can be associated with multiple RDS instances.
- Too many security group rules will increase the first packet latency. You are advised to create no more than 50 rules for a security group.
- To enable access to an RDS DB instance from resources outside the security group, you need to configure an **inbound rule** for the security group associated with the RDS DB instance.

NOTE

To ensure the security of your data and DB instances, you are advised to use the principle of least privilege for database access. Change the database port (default value: **3306**), and set the IP address to the remote server's address or any IP address in the remote server's smallest subnet to control the access from the remote server.

The default value of **Source** is **0.0.0.0/0**, indicating that RDS DB instances in the security group can be accessed from any IP address.

Procedure

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

Step 2 Click in the upper left corner and select a region and a project.

Step 3 Click in the upper left corner of the page and choose **Databases** > **Relational Database Service**.

Step 4 On the **Instances** page, click the DB instance name.

Step 5 In the navigation pane on the left, choose **Connectivity & Security**. In the **Security Group Rules** area, view security group rules.

Step 6 Click **Add Inbound Rule** or **Allow All IP** to configure security group rules.

To add more inbound rules, click .

NOTE

Allow All IP allows all IP addresses to access RDS DB instances in the security group, which poses high security risks. Exercise caution when performing this operation.

Figure 2-10 Adding an inbound rule

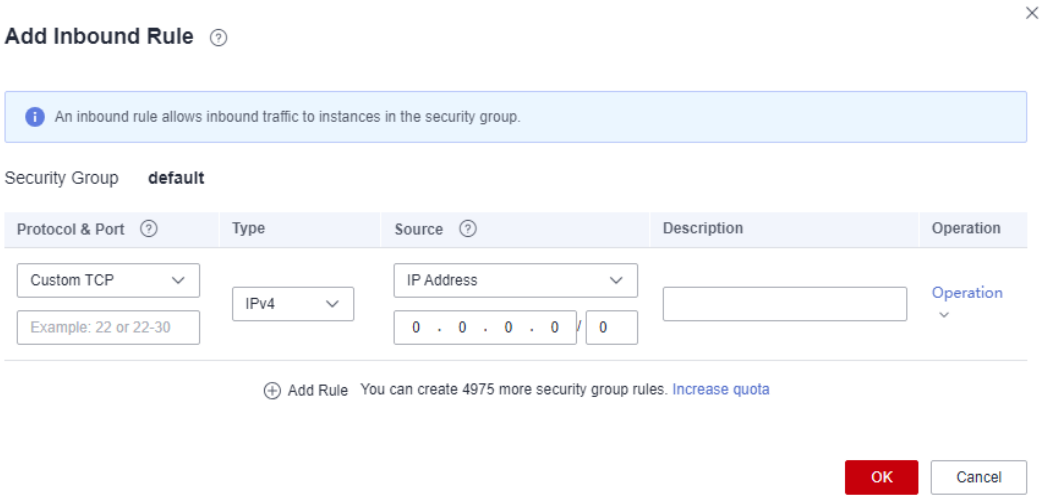


Table 2-10 Inbound rule parameter description

Parameter	Description	Example Value
Protocol & Port	**Protocol**: network protocol. Available options: **All ports**, **Custom TCP**, **Custom UDP**, **ICMP**, or **GRE**.	Custom TCP
	Port: the port over which the traffic can reach your DB instance. RDS for MariaDB instances can use database ports 1024 to 65535, excluding 12017 and 33071, which are reserved for RDS system use.	3306
Type	Supported source IP address type. Its value can be: • IPv4 • IPv6	IPv4

Parameter	Description	Example Value
Source	The source in an inbound rule is used to match the IP address or address range of an external request. The source can be: • Single IP address: 192.168.10.10/32 (IPv4 address) • IP address segment: 192.168.1.0/24 (IPv4 address segment) • All IP addresses: 0.0.0.0/0 (any IPv4 address) • Security group: sg-abc • IP address group: ipGroup-test	0.0.0.0/0
Description	Supplementary information about the security group rule. This parameter is optional. The description can contain a maximum of 255 characters and cannot contain angle brackets (<) or (>).	N/A

Step 7 Click **OK**.

----End

2.3.3.4 Connecting to a DB Instance Using a MariaDB Client

You can connect to an instance through a non-SSL connection or an SSL connection using a MariaDB client. SSL encrypts connections to your DB instance and is more secure.

Prerequisites

1. An EIP has been bound to the target DB instance and security group rules have been configured. The operations are as follows:
   a. Bind an EIP to your DB instance.
      For details about how to bind an EIP, see **Binding an EIP**.
   b. Obtain the IP address of the ECS you use to connect to the DB instance.
   c. Configure security group rules.
      Add the IP address obtained in **1.b** and the DB instance port to the inbound rule of the security group.
      For details about how to configure a security group rule, see **Configuring Security Group Rules**.

d. Run the **ping** command to check the connectivity between the ECS and the EIP that has been bound to the DB instance in **1.a**.

2. You have installed a database client to connect to DB instances.

   You can use a database client to connect to the target DB instance in Linux or Windows.

   - In Linux, you need to install a **MariaDB client** on your device. It is recommended that you download a MariaDB client running a version later than that of the DB instance.
   - In Windows, you can use any common database client to connect to the target DB instance in a similar way.

Connecting to a DB Instance Using Commands (SSL Connection)

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

Step 2 Click in the upper left corner and select a region and a project.

Step 3 Click in the upper left corner of the page and choose **Databases** > **Relational Database Service**.

Step 4 On the **Instances** page, click the instance name to go to the **Basic Information** page.

Step 5 In the **DB Information** area, check whether SSL is enabled.

- If yes, go to **6**.
- If no, click . In the displayed dialog box, click **OK**. Then, go to **6**.

Step 6 Click next to the **SSL** field to download **Certificate Download.zip**, and extract the root certificate **ca.pem** and bundle **ca-bundle.pem** from the package.

Step 7 Import the root certificate **ca.pem** to the Linux or Windows. For details, see **How Can I Import the Root Certificate to a Windows or Linux OS?**

Step 8 Connect to the RDS for MariaDB instance. In Linux, for example, run the following command:

mysql -h <*host*> **-P** <*port*> **-u** <*userName*> **-p --ssl-ca=**<*caName*>

Example:

mysql -h 172.16.0.31 -P 3306-u root -p --ssl-ca=ca.pem

Table 2-11 Parameter description

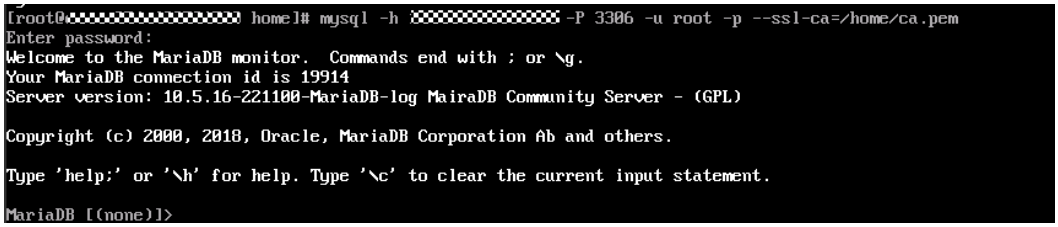
Parameter	Description
<*host*>	EIP of the DB instance to be connected.
<*port*>	Port of the DB instance to be connected.
<*userName*>	Database account used for logging in to the DB instance. The default value is **root**.

Parameter	Description
<caName>	Name of the CA certificate. The certificate should be stored in the directory where the command is executed.

Step 9 Enter the password of the database account if the following information is displayed:

```
Enter password:
```

Figure 2-11 Connection example



NOTE

If the connection fails, ensure that preparations have been correctly made in **Prerequisites** and try again.

----End

2.3.4 Connecting to a DB Instance Through DAS

Scenarios

Data Admin Service (DAS) enables you to connect to and manage DB instances with ease on a web-based console. The permission required for connecting to DB instances through DAS has been enabled for you by default. Using DAS to connect to your DB instance is recommended, which is more secure and convenient.

Procedure

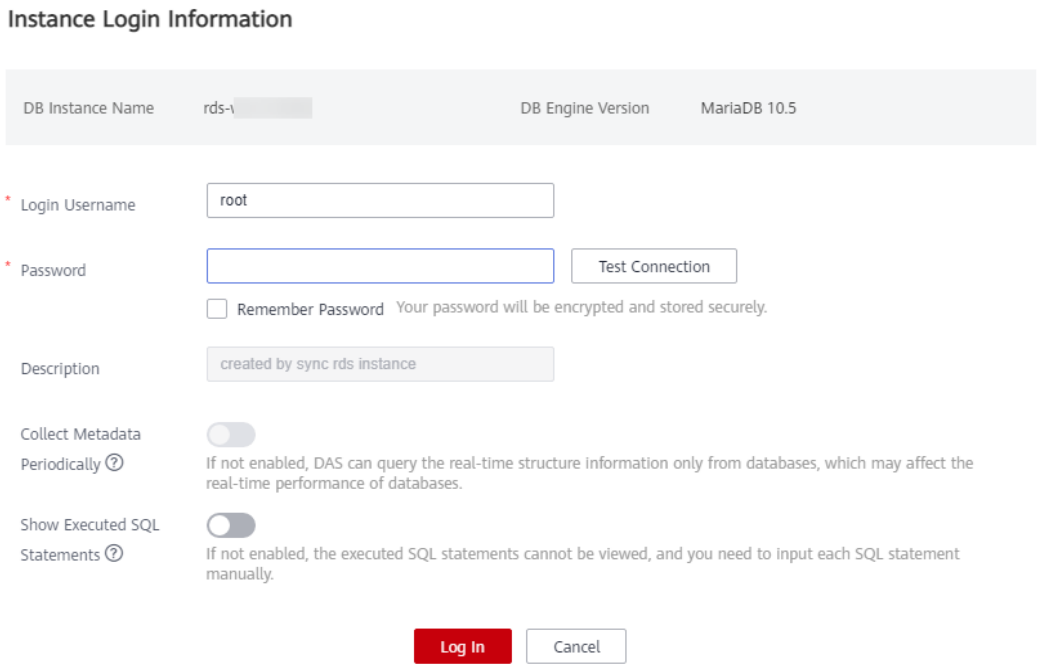
Step 1 **Log in to the management console**.

Step 2 Click in the upper left corner and select a region and a project.

Step 3 Click in the upper left corner of the page and choose **Databases** > **Relational Database Service**.

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

Figure 2-12 Login page



Step 5 Enter the database username and password and click **Test Connection**.

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

For details about how to manage databases using DAS, see **RDS for MariaDB Database Management** in the *Data Admin Service User Guide*.

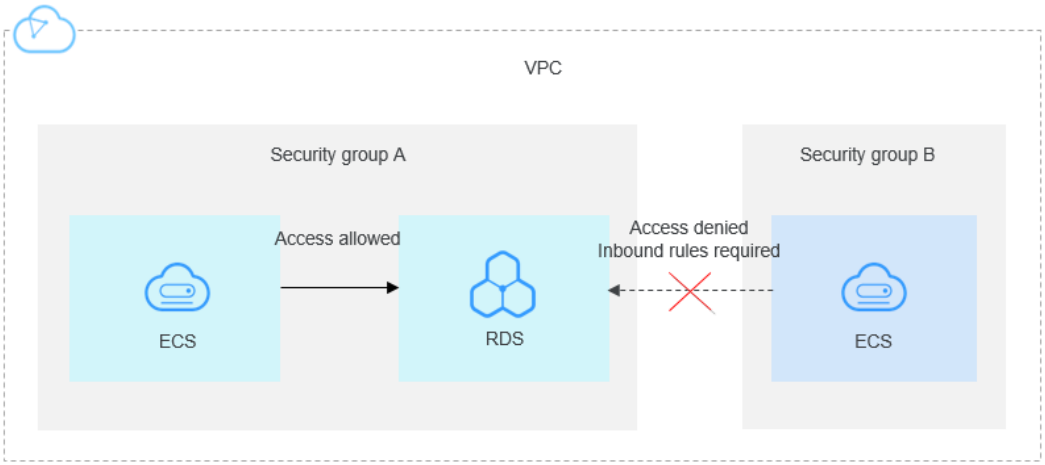
----End

2.4 Example: Buy and Connect to an RDS for MariaDB Instance

This example illustrates how to purchase an RDS for MariaDB instance and connect to it from a Linux ECS over a private network.

- **Step 1: Buy an RDS for MariaDB Instance**
- **Step 2: Buy an ECS**
- **Step 3: Connect to the RDS for MariaDB Instance**

Figure 2-13 Example diagram



Step 1: Buy an RDS for MariaDB Instance

1. Go to the **Buy DB Instance** page.
2. Configure the instance information and click **Next**. Keep the region, AZ, VPC, and security group of the DB instance the same as those of the ECS.

Figure 2-14 Selecting an engine version

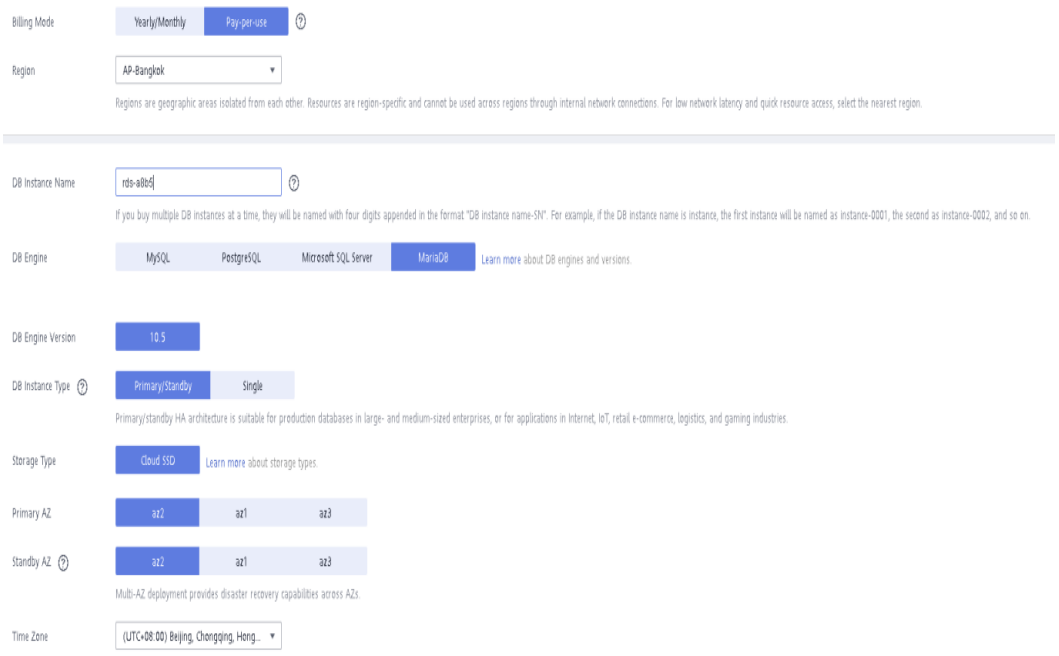


Figure 2-15 Selecting an instance class

Figure 2-16 Configuring network information

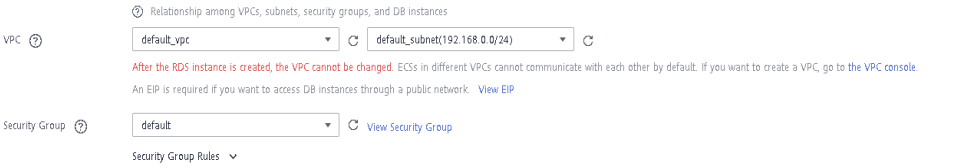
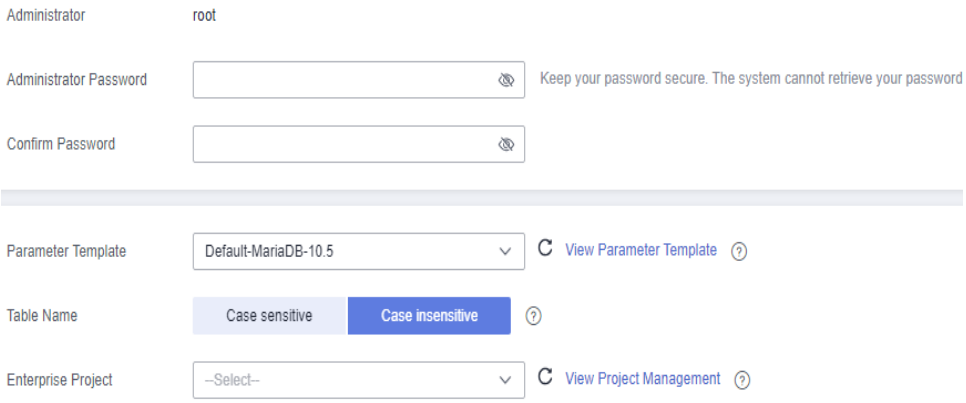
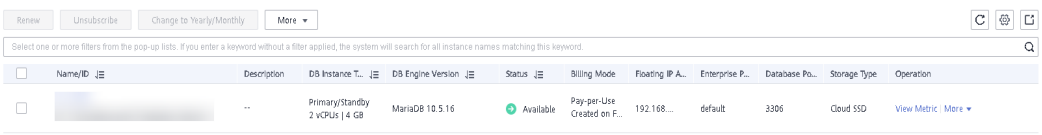


Figure 2-17 Setting a password



3. View the purchased RDS instance.

Figure 2-18 Instance successfully purchased



Step 2: Buy an ECS

1. Go to the **Buy ECS** page.
2. Configure basic settings and click **Next: Configure Network**. Keep the region and AZ of the ECS the same as those of the RDS for MariaDB instance to be connected.

Figure 2-19 Basic configurations

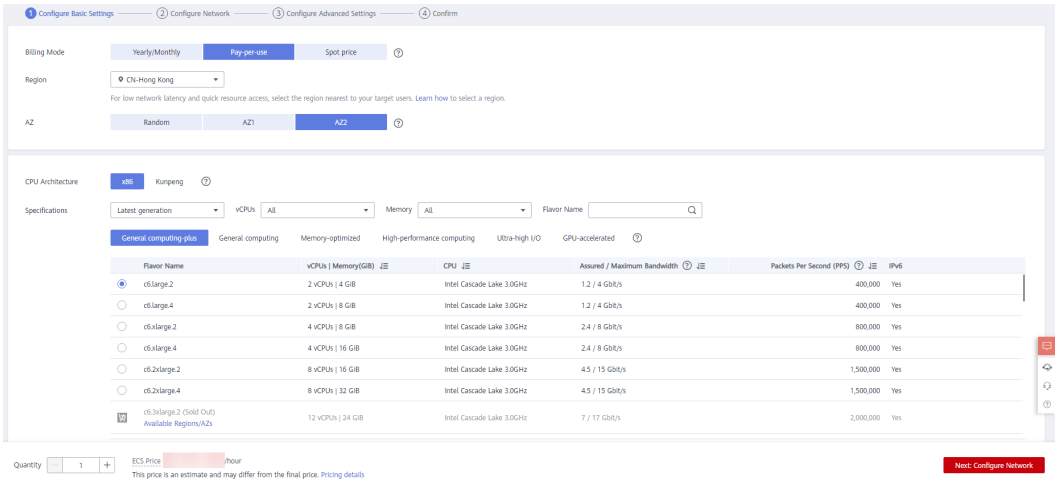
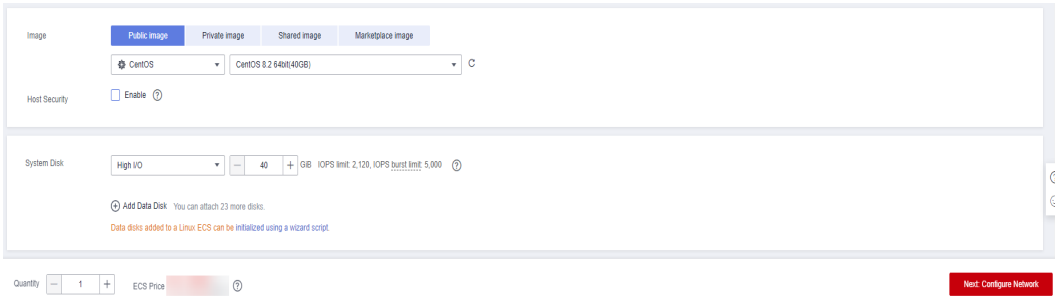


Figure 2-20 Selecting an image



3. Configure the ECS network information and click **Next: Configure Advanced Settings**. Keep the VPC and security group of the ECS the same as those of the RDS for MariaDB instance to be connected.

Figure 2-21 Network settings

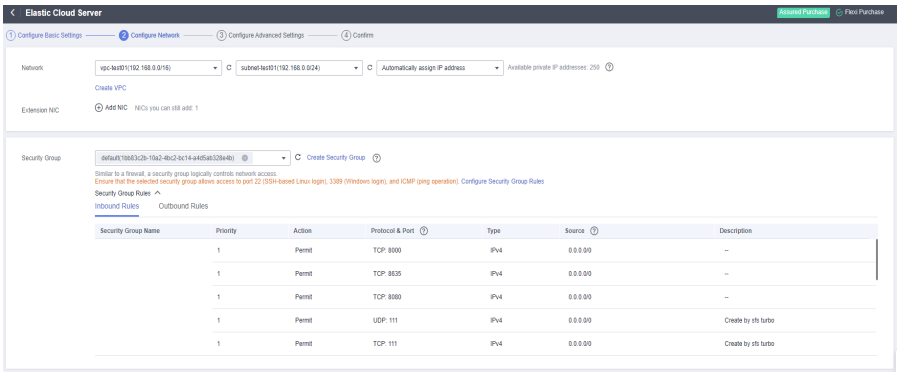
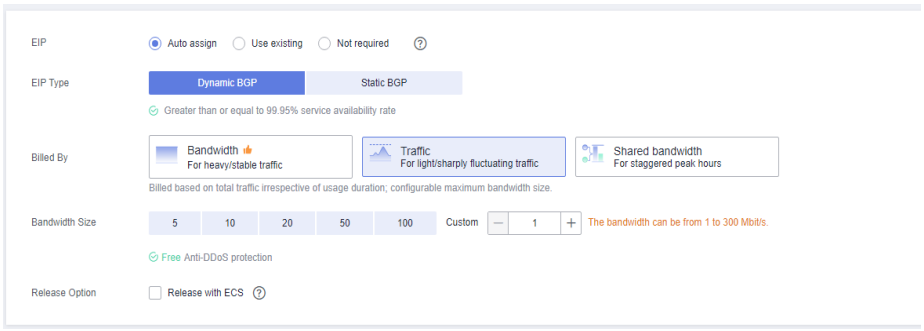
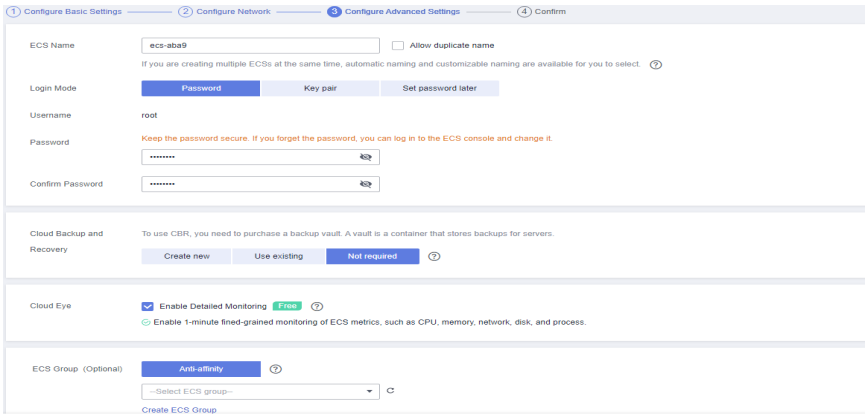


Figure 2-22 Selecting an EIP



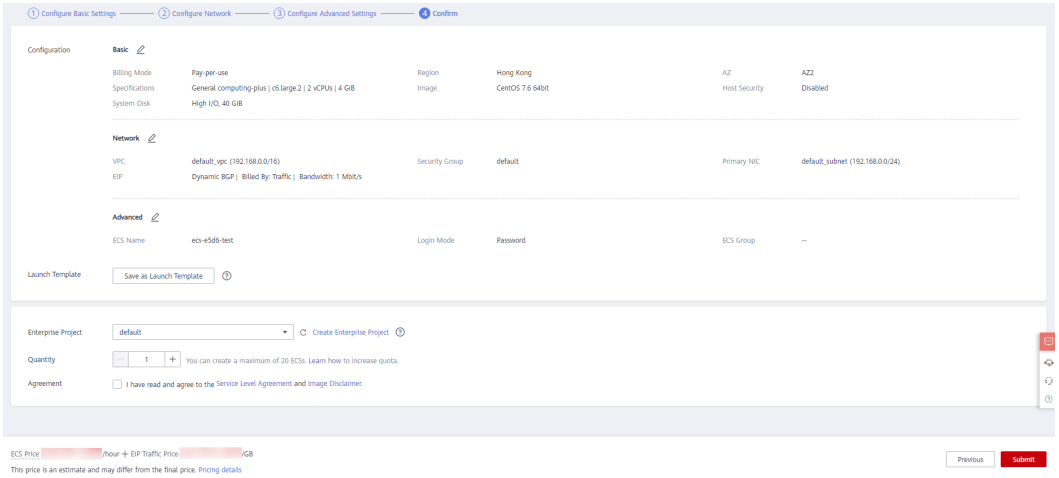
4. Configure the ECS password and click **Next: Confirm**.

Figure 2-23 Advanced settings



5. Confirm the configurations and click **Submit**.

Figure 2-24 Confirming the configurations

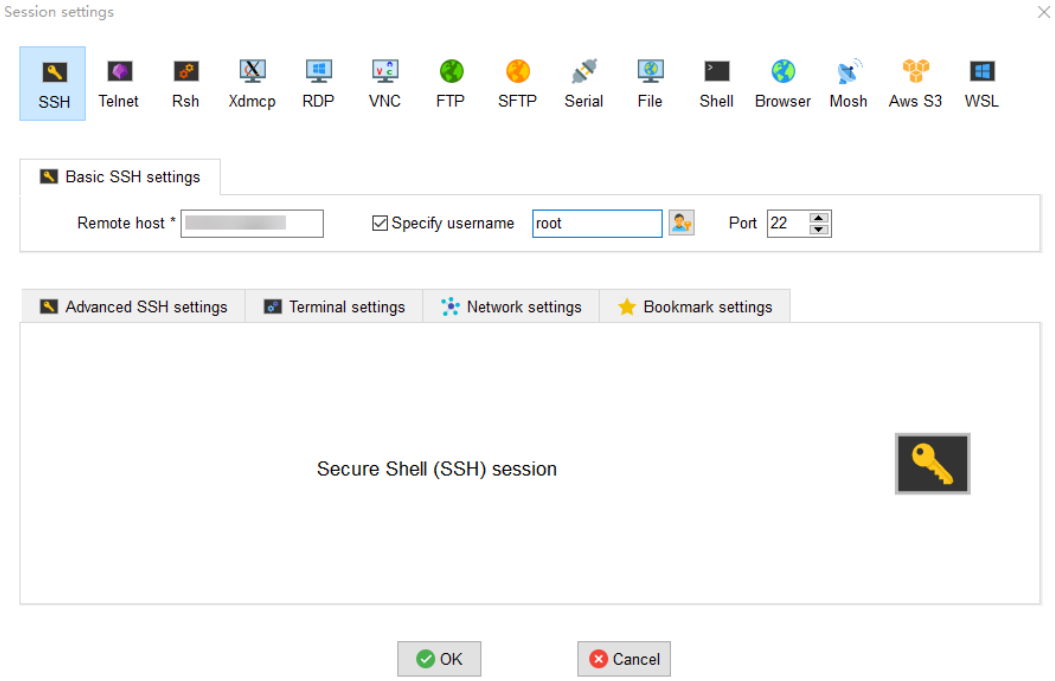


6. View the purchased ECS.

Step 3: Connect to the RDS for MariaDB Instance

1. Use a Linux remote connection tool (for example, MobaXterm) to log in to the ECS. Enter the EIP bound to the ECS for **Remote host**.

Figure 2-25 Creating a session



2. Enter the password of the ECS.

Figure 2-26 Entering the password

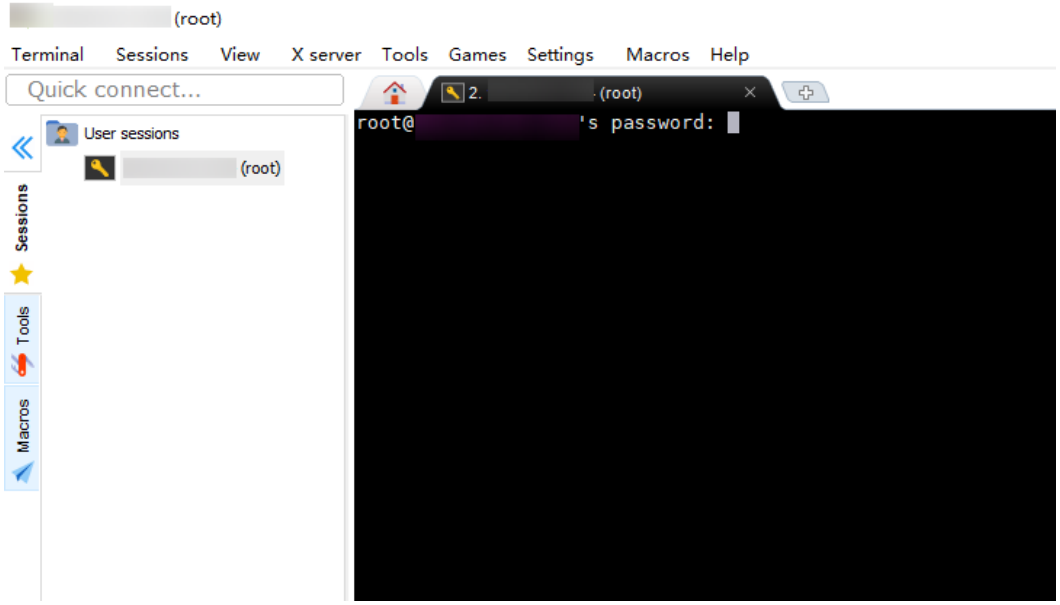
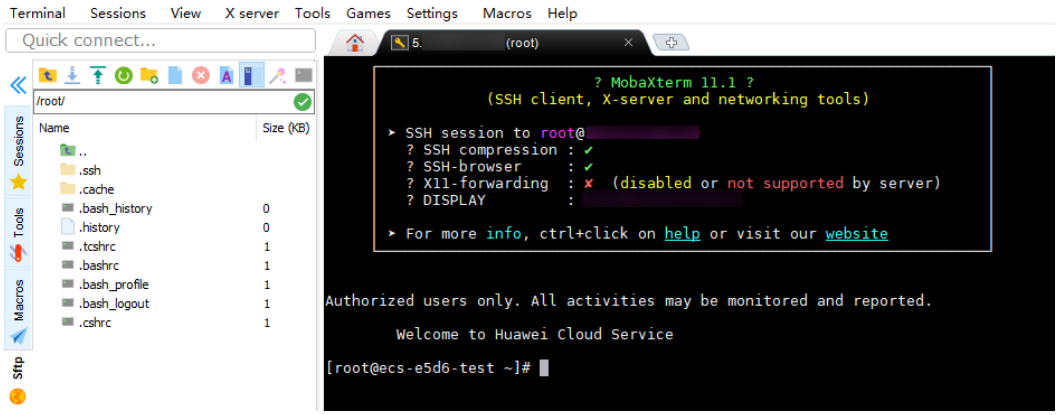


Figure 2-27 Successful login

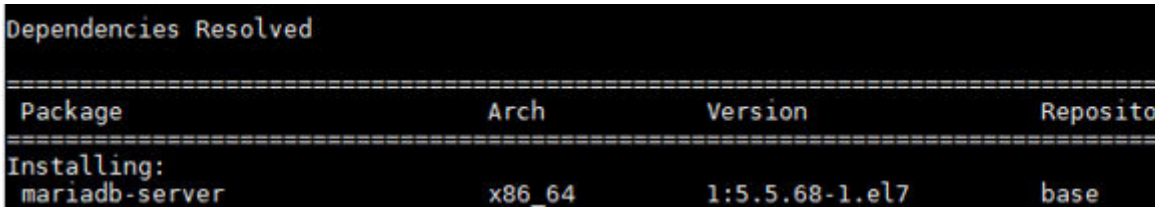


3. Install a **MariaDB client** by following the instructions provided in the official documentation.

   In CentOS, for example, run the following statement:

   ```
   yum install MariaDB-client
   ```

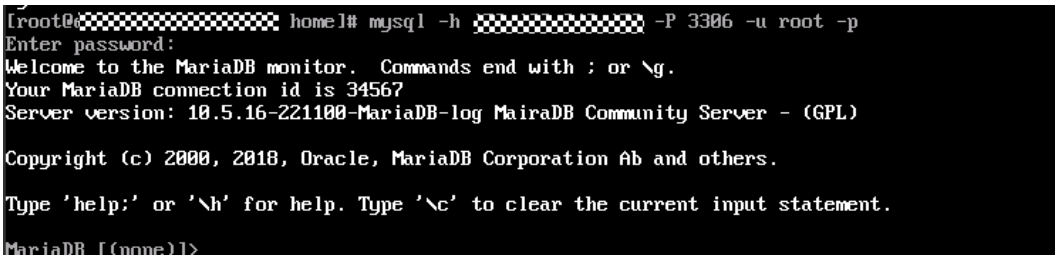
Figure 2-28 Installing a client



4. Connect to the RDS for MariaDB instance.

   ```
   mysql -h ip -P 3306 -u root -p
   ```

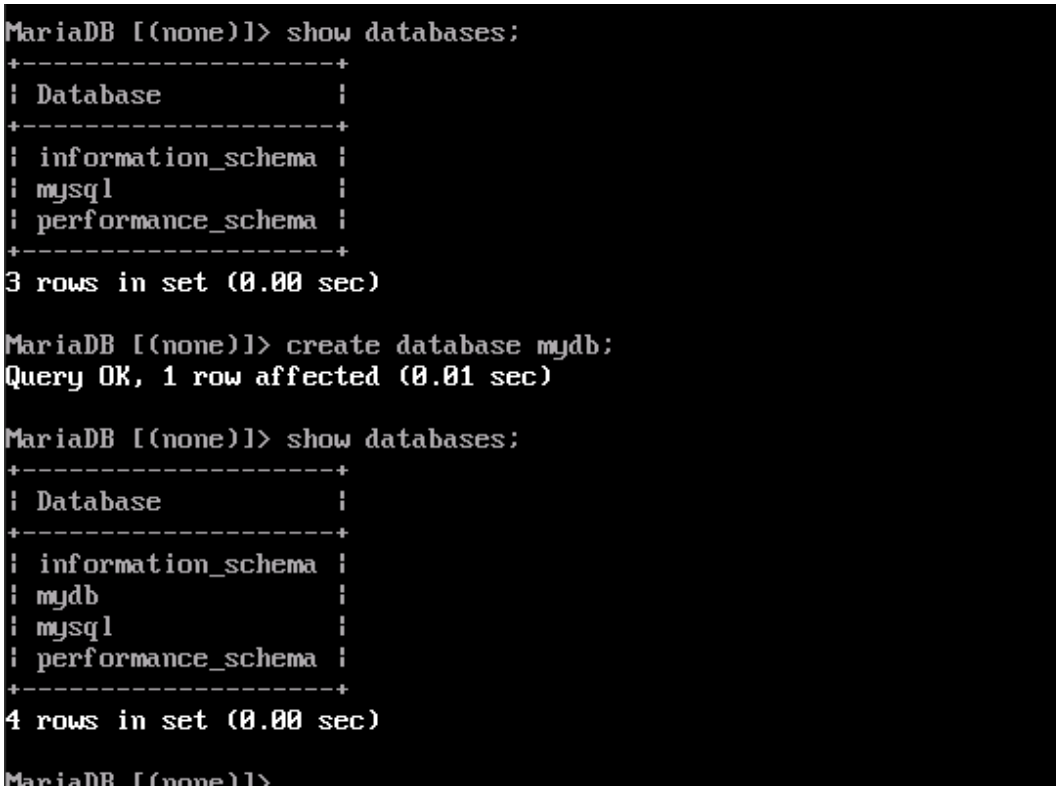
Figure 2-29 Connection succeeded



5. Create a database, for example, **mydb**.

   ```
   create database mydb;
   ```

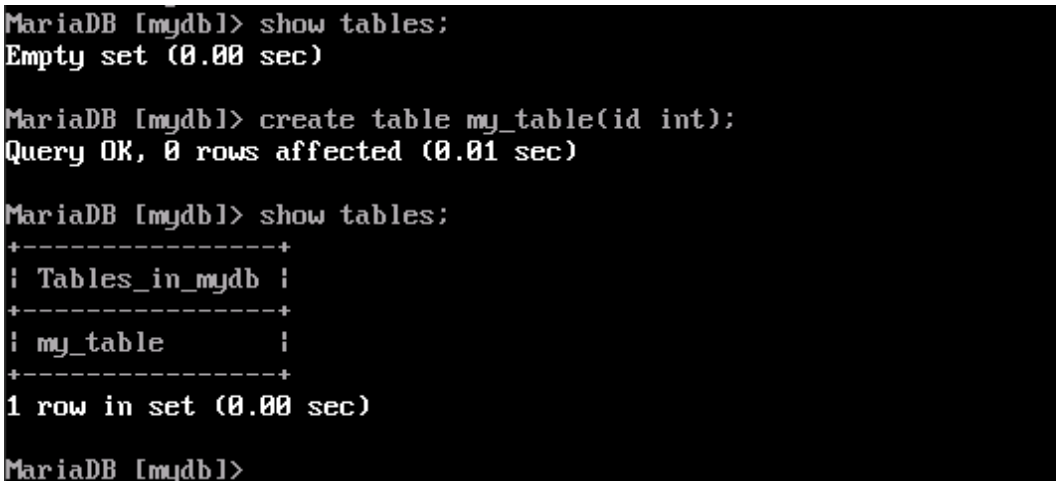

Figure 2-30 Creating a database



6. Create a table, for example, **my_table**.

```
create table my_table(id int);
```

Figure 2-31 Creating a table



3 Getting Started with RDS for PostgreSQL

3.1 Buying a DB Instance and Connecting to It Using the PostgreSQL Client

You can connect to your DB instance using a Linux ECS installed with the PostgreSQL client over a private network.

You can use the PostgreSQL client to connect to your DB instance over a Secure Sockets Layer (SSL) connection. SSL encrypts connections to your DB instance, making in-transit data more secure.

SSL is enabled by default when you create an RDS for PostgreSQL DB instance and cannot be disabled after the instance is created.

Enabling SSL reduces the read-only and read/write performance of your instance by about 20%.

Operation Process

Process	Description
Preparations	Sign up for a HUAWEI ID, enable Huawei Cloud services, make sure you have a valid payment method configured, create IAM users, and grant them specific RDS permissions.
Step 1: Buy an RDS for PostgreSQL DB Instance	Select required basic settings and additional options and buy an RDS for PostgreSQL DB instance.

Process	Description
Step 2: Buy an ECS	If you want to use the PostgreSQL client to connect to a DB instance, you need to prepare a server, install the PostgreSQL client on the server, and run the connection command. Purchase a Linux ECS that is in the same region and VPC as your DB instance.
Step 3: Test Connectivity and Install the PostgreSQL Client	Test the network connectivity between the ECS and the floating IP address and port of the DB instance, and install the PostgreSQL client on the ECS.
Step 4: Connect to the DB Instance Using Commands (SSL Connection)	Use a command-line interface (CLI) to connect to the DB instance using the floating IP address and port.

Preparations

1. **Sign up for a HUAWEI ID and enable Huawei Cloud services**.
2. Before purchasing DB instances, ensure that your account balance is sufficient. **Top up your account** if required.
3. For fine-grained permissions management on Huawei Cloud resources, use Identity and Access Management (IAM) to create a user or user group and grant it specific operation permissions. For details, see **Creating a User and Granting Permissions**.

Step 1: Buy an RDS for PostgreSQL DB Instance

1. Go to the **Buy DB Instance** page.
2. On the **Quick Config** page, set basic parameters.

   NOTE

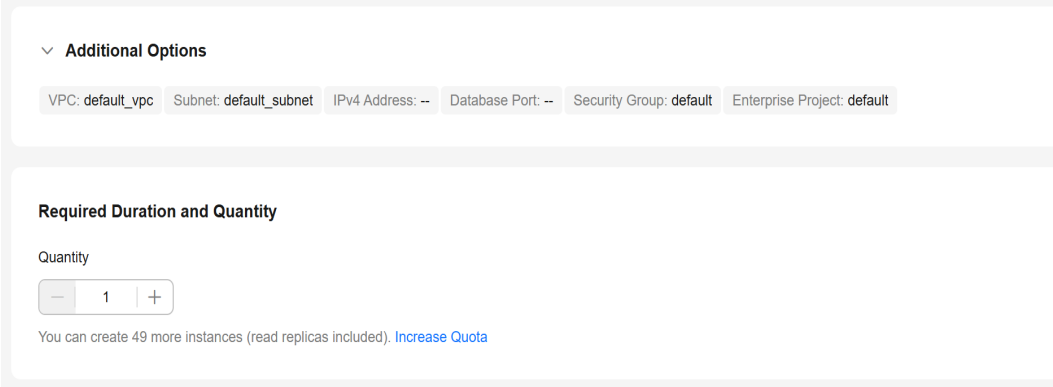
   The following parameter settings are only for reference.

Parameter	Example Value	Description
Billing Mode	Pay-per-use	The billing mode of an instance. • **Yearly/Monthly**: A prepaid billing mode in which you pay for resources before using it. Bills are settled based on the subscription period. The longer the subscription, the bigger the discount. This mode is a good option for long-term, stable services. • **Pay-per-use**: A postpaid billing mode. You pay as you go and just pay for what you use. The DB instance usage is calculated by the second but billed every hour. This mode allows you to adjust resource usage easily. You neither need to prepare for resources in advance, nor end up with excessive or insufficient preset resources.
Region	CN-Hong Kong	The region where your resources are located. **NOTE** Products in different regions cannot communicate with each other through a private network. After a DB instance is created, the region cannot be changed. Therefore, exercise caution when selecting a region.
DB Engine Version	16	The database version.
DB Instance Type	Primary/ Standby	The architecture type of an instance. **Primary/Standby**: An HA architecture. In a primary/standby pair, each instance has the same instance class. When a primary instance is being created, a standby instance is provisioned along with it to provide data redundancy. The standby instance is invisible to you after being created.
Instance Class	General-purpose | 4U | 8G	The vCPU and memory of an instance.
Storage	Cloud SSD | 100GB	The storage space of an instance. It contains the system overhead required for inodes, reserved blocks, and database operation.

Parameter	Example Value	Description
Disk Encryption	Disable	Enabling disk encryption improves data security, but slightly affects the read and write performance of the database. If a shared KMS key is used, the corresponding CTS events are **createdatakey** and **decrydatakey**. Only the key owner can receive the events.

3. Complete advanced configurations.

Figure 3-1 Additional Options

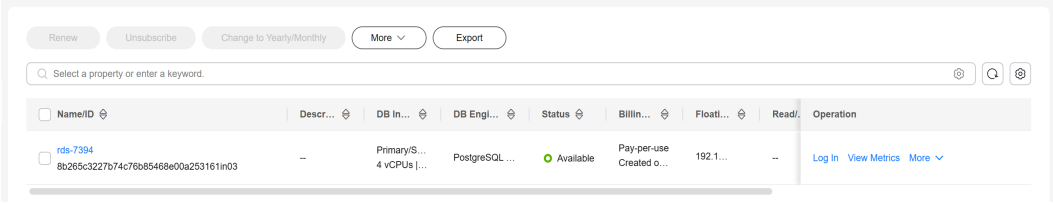


Parameter	Example Value	Description
VPC	default_vpc	The virtual network in which your instance is located. A VPC can isolate networks for different workloads. If no VPC is available, click **Create VPC**. After a VPC is created, click . For details, see **Creating a VPC and Subnet**.
Subnet	default_subnet	A subnet provides dedicated network resources that are logically isolated from other networks for network security.
Security Group	default	It can enhance security by controlling access to RDS for PostgreSQL from other services.
Enterprise Project	default	If your account has been associated with an enterprise project, select the target project from the **Enterprise Project** drop-down list. For more information about enterprise projects, see ***Enterprise Management User Guide***.

Parameter	Example Value	Description
Quantity	1	The number of instances to be created in a batch.

4. Click **Buy**.
5. Check the purchased DB instance.

Figure 3-2 Instance successfully purchased



Step 2: Buy an ECS

1. Go to the **Elastic Cloud Server console**.
2. Check whether there is a Linux ECS that meets the requirements.

NOTICE

RDS for PostgreSQL supports the following client installation methods:

- Download the PostgreSQL client installation package. This method is recommended for PostgreSQL 15 and earlier versions. **It has requirements on ECS images. For details, see the official PostgreSQL documentation.**
- Download the source code. This method has no requirements on PostgreSQL versions or ECS images.

- If yes, go to **3**.
- If no, purchase an ECS and select Linux (for example, CentOS 7) as its OS.

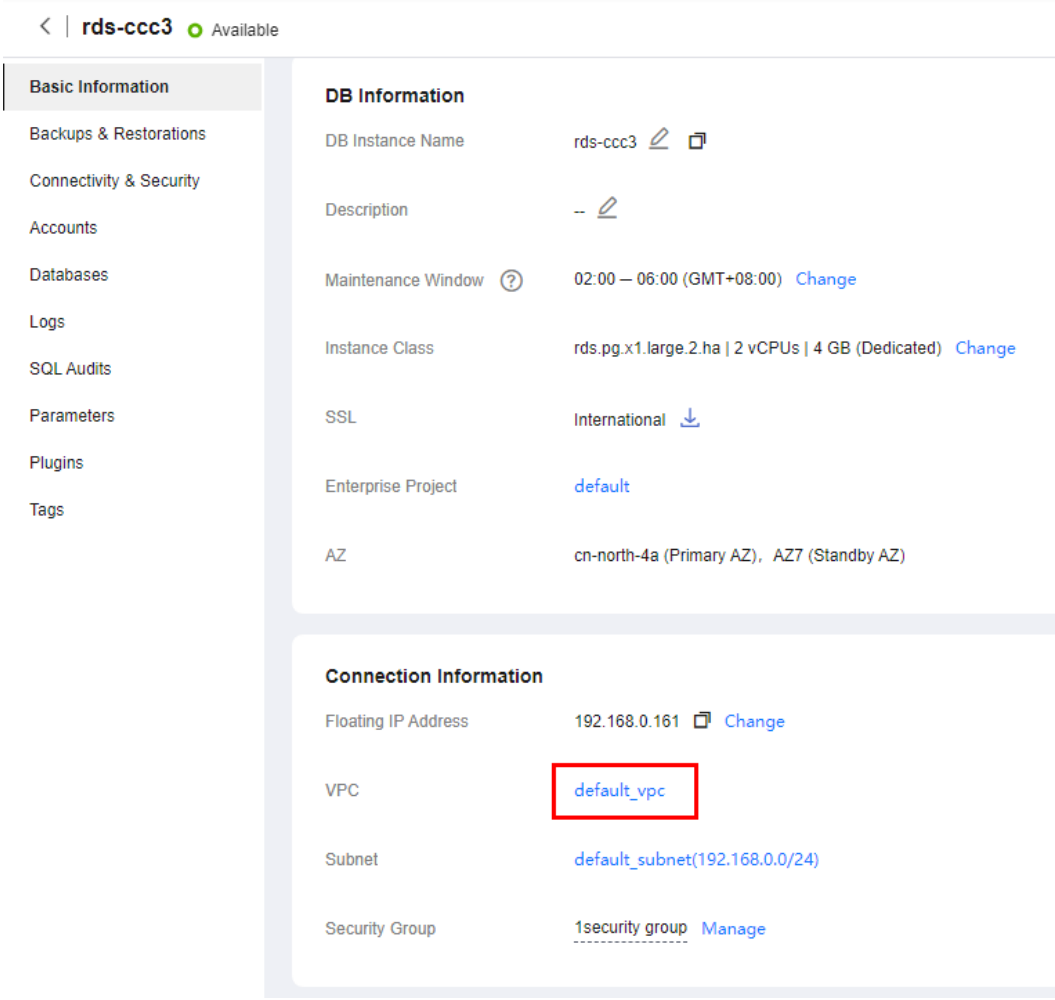
  To download the PostgreSQL client to the ECS, bind an EIP to the ECS. The ECS must be in the same region, VPC, and security group as the RDS for PostgreSQL DB instance for mutual communications.

  For details about how to purchase a Linux ECS, see **Purchasing a Custom ECS** in *Elastic Cloud Server User Guide*.

3. Check whether the ECS and RDS for PostgreSQL instance are in the same region and VPC.

Figure 3-3 ECS information

Figure 3-4 Overview



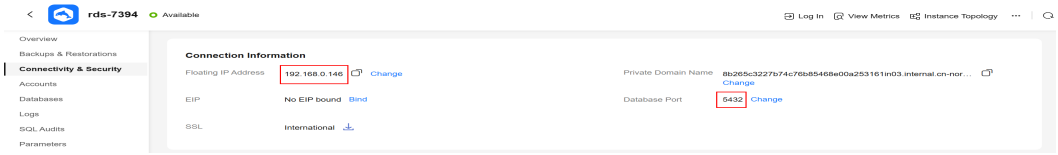
- If they are not in the same region, purchase another ECS. The ECS and DB instance in different regions cannot communicate with each other. To reduce network latency, deploy your DB instance in the region nearest to your workloads.
- If the ECS and DB instance are in different VPCs, change the VPC of the ECS to that of the DB instance. For details, see **Changing a VPC**.

Step 3: Test Connectivity and Install the PostgreSQL Client

Installing the PostgreSQL Client (PostgreSQL 15 and Earlier)

1. Log in to the ECS. For details, see **Login Using VNC** in the *Elastic Cloud Server User Guide*.
2. On the **Instances** page of the RDS console, click the DB instance name.
3. Choose **Connectivity & Security** from the navigation pane. In the **Connection Information** area, obtain the floating IP address and database port of the DB instance.

   Figure 3-5 Connection information

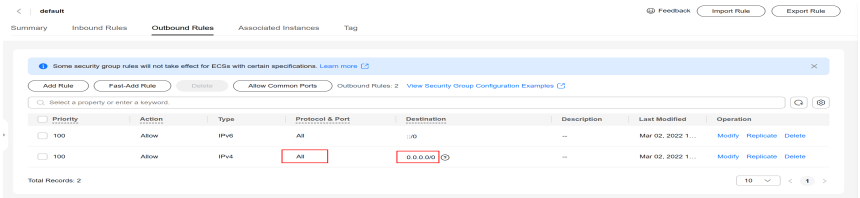
4. On the ECS, check whether the floating IP address and database port of the DB instance can be connected.

   curl -kv *192.168.0.7*:*5432*

   - If yes, network connectivity is normal.
   - If no, check the security group rules.

     - If in the security group of the ECS, there is no outbound rule with **Destination** set to **0.0.0.0/0** and **Protocol & Port** set to **All**, add an outbound rule for the floating IP address and port of the DB instance.

       Figure 3-6 ECS security group

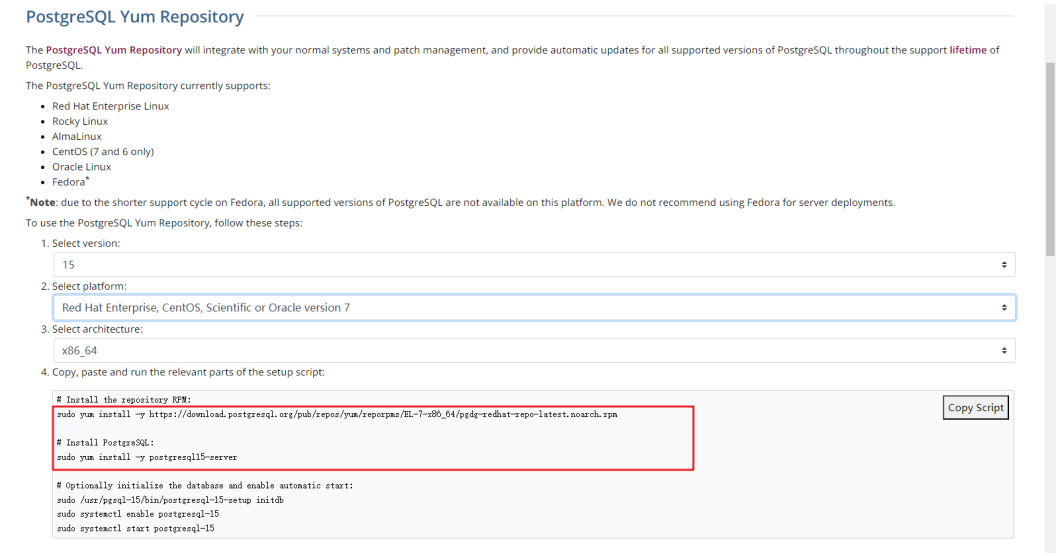
     - If in the security group of the DB instance, there is no inbound rule allowing the access from the private IP address and port of the ECS, add an inbound rule for the private IP address and port of the ECS.

5. Install the PostgreSQL client.

   The PostgreSQL community provides **client installation methods** for different OSs. You can download and install the client using the installation tool of the OS. This installation method is simple but can be used only for the OSs supported by the PostgreSQL community.

In this example, CentOS 7 is used. Use the default installation tool of the OS to install the client (PostgreSQL 15 or earlier).

Figure 3-7 Obtaining the installation tool



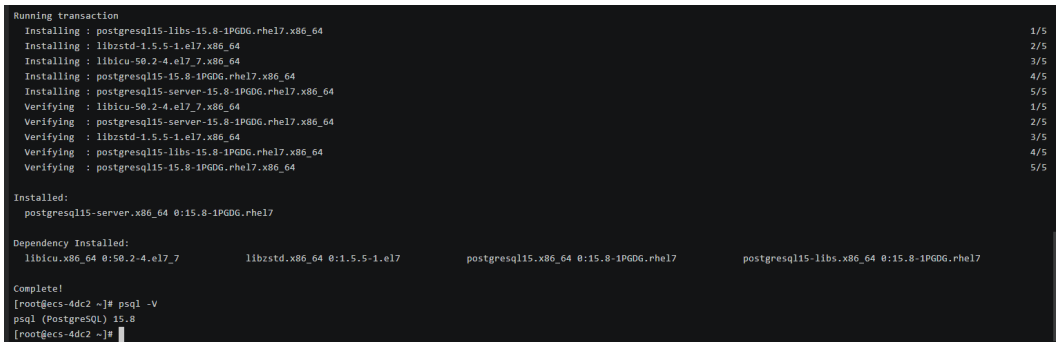
Run the following commands:

```
sudo yum install -y https://download.postgresql.org/pub/repos/yum/reporpms/EL-7-x86_64/pgdg-redhat-repo-latest.noarch.rpm
sudo yum install -y postgresql15-server
```

Check whether the installation is successful.

```
psql -V
```

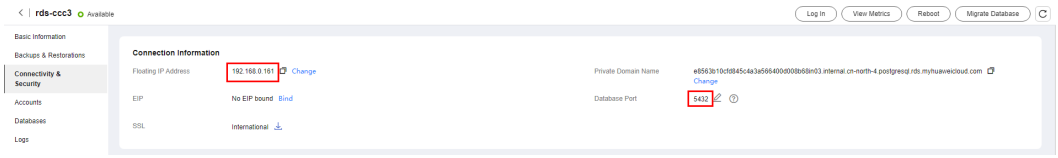
Figure 3-8 Successful installation



Installing the PostgreSQL Client (No Restrictions on PostgreSQL Versions)

1. Log in to the ECS. For details, see **Login Using VNC** in the *Elastic Cloud Server User Guide*.
2. On the **Instances** page of the RDS console, click the DB instance name.
3. Choose **Connectivity & Security** from the navigation pane. In the **Connection Information** area, obtain the floating IP address and database port of the DB instance.

Figure 3-9 Connection information

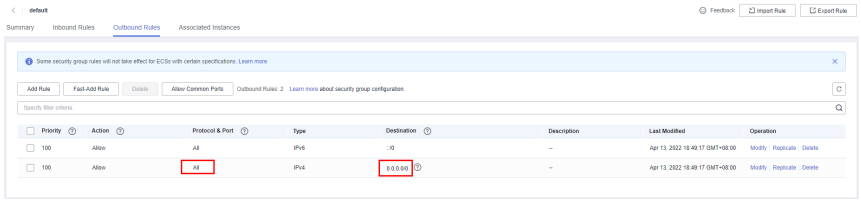


4. On the ECS, check whether the floating IP address and database port of the DB instance can be connected.

```
curl -kv 192.168.0.7:5432
```

   – If yes, network connectivity is normal.
   – If no, check the security group rules.
      ■ If in the security group of the ECS, there is no outbound rule with **Destination** set to **0.0.0.0/0** and **Protocol & Port** set to **All**, add an outbound rule for the floating IP address and port of the DB instance.

        Figure 3-10 ECS security group

      ■ If in the security group of the DB instance, there is no inbound rule allowing the access from the private IP address and port of the ECS, add an inbound rule for the private IP address and port of the ECS.

5. Install the PostgreSQL client.

   Installation from source code: This installation method has no restrictions on RDS PostgreSQL instance versions and ECS OS types.

   The following uses an ECS using the Huawei Cloud EulerOS 2.0 image as an example to describe how to install a PostgreSQL 16.4 client.

Figure 3-11 Checking the ECS image

   a. To use SSL, download OpenSSL to the ECS in advance.

```
sudo yum install -y openssl-devel
```


b. Obtain the **code download link**, run **wget** to download the installation package to the ECS, or download the installation package to the local PC and upload it to the ECS.

```
wget https://ftp.postgresql.org/pub/source/v16.4/postgresql-16.4.tar.gz
```

c. Decompress the installation package.

```
tar xf postgresql-16.4.tar.gz
```

d. Compile the source code and then install the client.

```
cd postgresql-16.4
./configure --without-icu --without-readline --without-zlib --with-openssl
make -j 8 && make install
```

NOTE

If **--prefix** is not specified, the default path is **/usr/local/pgsql**. The client can be installed in the simplest way.

Figure 3-12 Compilation and installation

```
make -C ../../../src/common all
make[4]: Entering directory '/root/postgresql-16.4/src/common'
make[4]: Nothing to be done for 'all'.
make[4]: Leaving directory '/root/postgresql-16.4/src/common'
make[3]: Leaving directory '/root/postgresql-16.4/src/interfaces/libpq'
make -C ../../../src/port all
make[3]: Entering directory '/root/postgresql-16.4/src/port'
make[3]: Nothing to be done for 'all'.
make[3]: Leaving directory '/root/postgresql-16.4/src/port'
make -C ../../../src/common all
make[3]: Entering directory '/root/postgresql-16.4/src/common'
make[3]: Nothing to be done for 'all'.
make[3]: Leaving directory '/root/postgresql-16.4/src/common'
/usr/bin/mkdir -p '/usr/local/pgsql/lib/pgxs/src/test/isolation'
/usr/bin/install -c  pg_isolation_regress '/usr/local/pgsql/lib/pgxs/src/test/isolation/pg_isolation_regress'
/usr/bin/install -c  isolationtester '/usr/local/pgsql/lib/pgxs/src/test/isolation/isolationtester'
make[2]: Leaving directory '/root/postgresql-16.4/src/test/isolation'
make -C test/perl install
make[2]: Entering directory '/root/postgresql-16.4/src/test/perl'
make[2]: Nothing to be done for 'install'.
make[2]: Leaving directory '/root/postgresql-16.4/src/test/perl'
/usr/bin/mkdir -p '/usr/local/pgsql/lib/pgxs/src'
/usr/bin/install -c -m 644 Makefile.global '/usr/local/pgsql/lib/pgxs/src/Makefile.global'
/usr/bin/install -c -m 644 Makefile.port '/usr/local/pgsql/lib/pgxs/src/Makefile.port'
/usr/bin/install -c -m 644 ./Makefile.shlib '/usr/local/pgsql/lib/pgxs/src/Makefile.shlib'
/usr/bin/install -c -m 644 ./nls-global.mk '/usr/local/pgsql/lib/pgxs/src/nls-global.mk'
make[1]: Leaving directory '/root/postgresql-16.4/src'
make -C config install
make[1]: Entering directory '/root/postgresql-16.4/config'
/usr/bin/mkdir -p '/usr/local/pgsql/lib/pgxs/config'
/usr/bin/install -c -m 755 ./install-sh '/usr/local/pgsql/lib/pgxs/config/install-sh'
/usr/bin/install -c -m 755 ./missing '/usr/local/pgsql/lib/pgxs/config/missing'
make[1]: Leaving directory '/root/postgresql-16.4/config'
```

e. Add the following code to the **/etc/profile** file to configure environment variables:

```
export PATH=/usr/local/pgsql/bin:$PATH
export LD_LIBRARY_PATH=/usr/local/pgsql/lib:$LD_LIBRARY_PATH
source /etc/profile
```

f. Test whether the psql is available.

```
psql -V
```

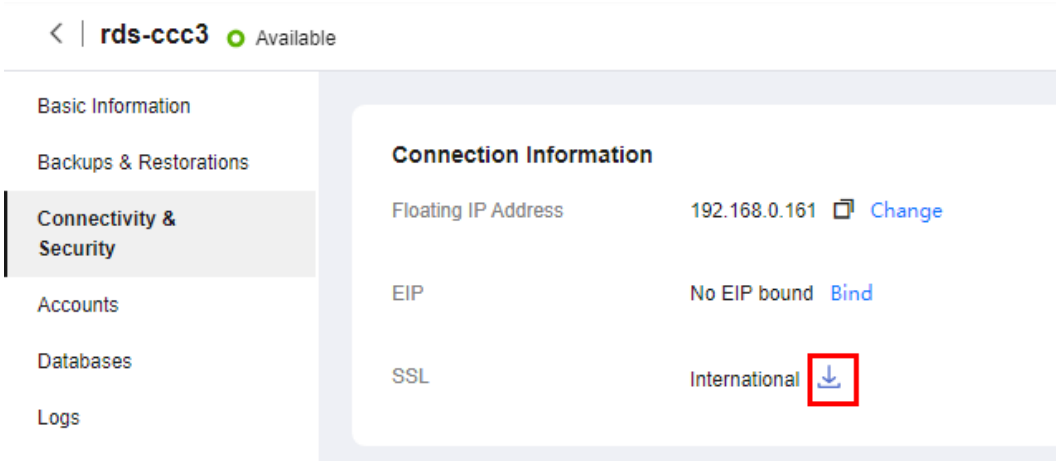

Figure 3-13 Testing psql

```
                . /etc/bashrc
        fi
fi
export PATH=/usr/local/pgsql/bin:$PATH
export LD_LIBRARY_PATH=/usr/local/pgsql/lib:$LD_LIBRARY_PATH
[root@ecs-88a7 pgsql]# source /etc/profile
[root@ecs-88a7 pgsql]# psql -V
psql (PostgreSQL) 16.4
[root@ecs-88a7 pgsql]#
```

Step 4: Connect to the DB Instance Using Commands (SSL Connection)

1. On the **Instances** page of the RDS console, click the DB instance name.
2. In the navigation pane, choose **Connectivity & Security**.
3. In the **Connection Information** area, click next to the **SSL** field to download **Certificate Download.zip**, and extract the root certificate **ca.pem** and bundle **ca-bundle.pem** from the package.

Figure 3-14 Downloading a certificate



4. Upload **ca.pem** to the ECS.

   NOTE

   - TLS v1.2 or later is recommended. Versions earlier than TLS v1.2 have security risks.
   - The recommended protocol algorithm is EECDH+ECDSA+AESGCM:EECDH+aRSA+AESGCM:EDH+aRSA+AESGCM:EDH+aDSS+AESGCM:!aNULL:!eNULL:!LOW:!3DES:!MD5:!EXP:!SRP:!RC4. Using other options have security risks.
   - **ca-bundle.pem** contains both the new certificate provided as of April 2017 and the old certificate.
   - Both **ca.pem** and **ca-bundle.pem** can be used for SSL connections because **ca-bundle.pem** contains **ca.pem**.

5. Run the following command on the ECS to connect to the DB instance:

   psql --no-readline -h *<host>* **-p** *<port>* **"dbname=***<database>* **user=***<user>* **sslmode=verify-ca sslrootcert=***<ca-file-directory>***"**

   Example:

   ```
   psql --no-readline -h 192.168.0.7 -p 5432 "dbname=postgres user=root sslmode=verify-ca sslrootcert=/root/ca.pem"
   ```


Table 3-1 Parameter description

Parameter	Description
<host>	Floating IP address obtained in 3.
<port>	Database port obtained in 3. The default value is **5432**.
<database>	Name of the database to be connected. The default database name is **postgres**.
<user>	Administrator account **root**.
<ca-file-directory>	Directory of the CA certificate used for the SSL connection. This certificate should be stored in the directory where the command is executed.
sslmode	SSL connection mode. Set it to **verify-ca** to use a CA to check whether the service is trusted.

6. Enter the password of the database account as prompted.

```
Password:
```

If the following information is displayed, the connection is successful.

```
SSL connection (protocol: TLSv1.2, cipher: ECDHE-RSA-AES256-GCM-SHA384, bits: 256, compression: off)
```

Follow-up Operations

After logging in to the DB instance, you can create or migrate databases.

- **Creating a PostgreSQL Database Using an API**
- **Managing PostgreSQL Databases Using DAS**
- **Migration Solution Overview**

3.2 Buying an RDS for PostgreSQL Instance and Connecting to It Through DAS

This section describes how to purchase an RDS for PostgreSQL instance and how to connect to it using DAS.

- **Step 1: Buy an RDS for PostgreSQL DB Instance**
- **Step 2: Connect to the RDS for PostgreSQL Instance**

Step 1: Buy an RDS for PostgreSQL DB Instance

1. Go to the **Buy DB Instance** page.
2. On the **Quick Config** page, set basic parameters.

NOTE

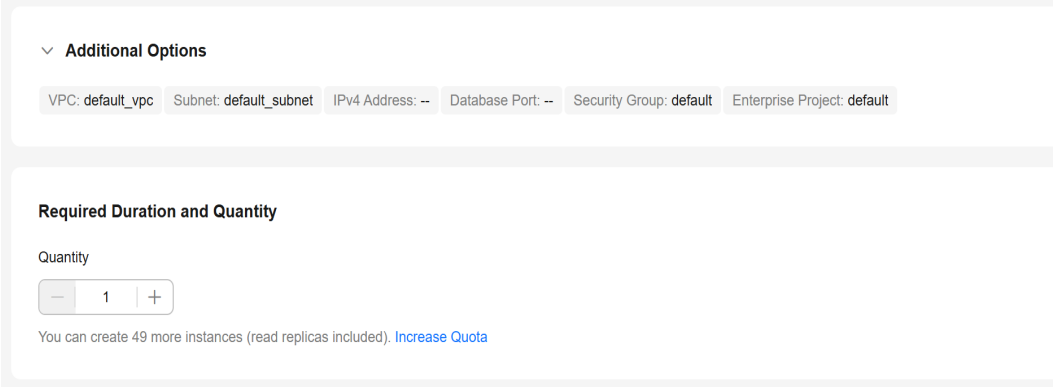
The following parameter settings are only for reference.

Parameter	Example Value	Description
Billing Mode	Pay-per-use	The billing mode of an instance. • **Yearly/Monthly**: A prepaid billing mode in which you pay for resources before using it. Bills are settled based on the subscription period. The longer the subscription, the bigger the discount. This mode is a good option for long-term, stable services. • **Pay-per-use**: A postpaid billing mode. You pay as you go and just pay for what you use. The DB instance usage is calculated by the second but billed every hour. This mode allows you to adjust resource usage easily. You neither need to prepare for resources in advance, nor end up with excessive or insufficient preset resources.
Region	CN-Hong Kong	The region where your resources are located. **NOTE** Products in different regions cannot communicate with each other through a private network. After a DB instance is created, the region cannot be changed. Therefore, exercise caution when selecting a region.
DB Engine Version	16	The database version.
DB Instance Type	Primary/ Standby	The architecture type of an instance. **Primary/Standby**: An HA architecture. In a primary/standby pair, each instance has the same instance class. When a primary instance is being created, a standby instance is provisioned along with it to provide data redundancy. The standby instance is invisible to you after being created.
Instance Class	General-purpose | 4U | 8G	The vCPU and memory of an instance.
Storage	Cloud SSD | 100GB	The storage space of an instance. It contains the system overhead required for inodes, reserved blocks, and database operation.

Parameter	Example Value	Description
Disk Encryption	Disable	Enabling disk encryption improves data security, but slightly affects the read and write performance of the database. If a shared KMS key is used, the corresponding CTS events are **createdatakey** and **decrydatakey**. Only the key owner can receive the events.

3. Complete advanced configurations.

Figure 3-15 Additional Options

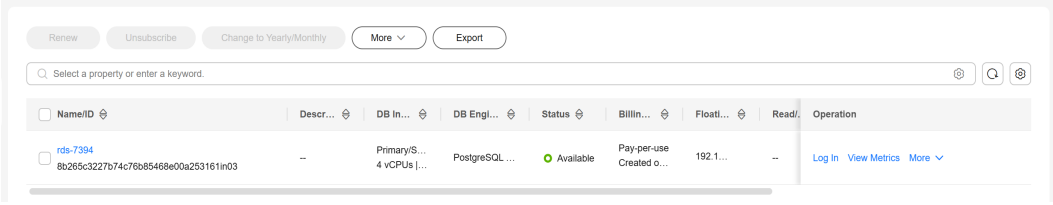


Parameter	Example Value	Description
VPC	default_vpc	The virtual network in which your instance is located. A VPC can isolate networks for different workloads. If no VPC is available, click **Create VPC**. After a VPC is created, click . For details, see **Creating a VPC and Subnet**.
Subnet	default_subnet	A subnet provides dedicated network resources that are logically isolated from other networks for network security.
Security Group	default	It can enhance security by controlling access to RDS for PostgreSQL from other services.
Enterprise Project	default	If your account has been associated with an enterprise project, select the target project from the **Enterprise Project** drop-down list. For more information about enterprise projects, see ***Enterprise Management User Guide***.

Parameter	Example Value	Description
Quantity	1	The number of instances to be created in a batch.

4. Click **Buy**.
5. Check the purchased DB instance.

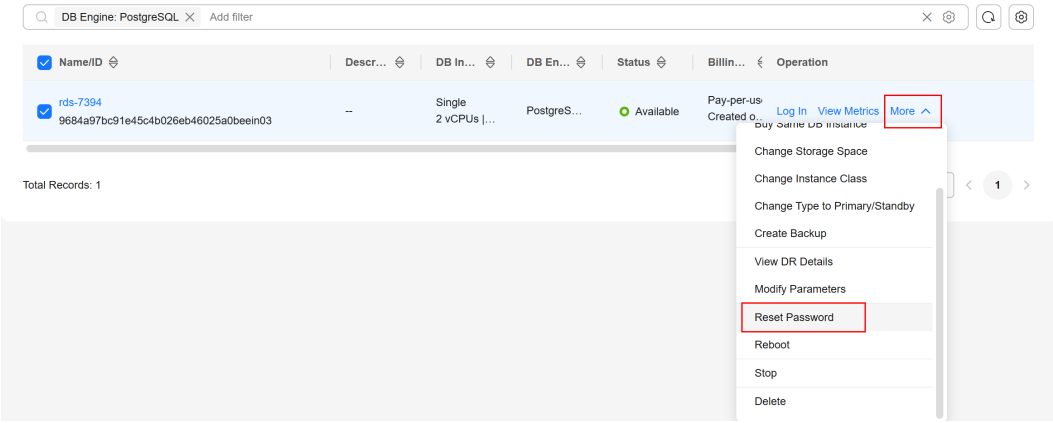
Figure 3-16 Instance successfully purchased



Step 2: Connect to the RDS for PostgreSQL Instance

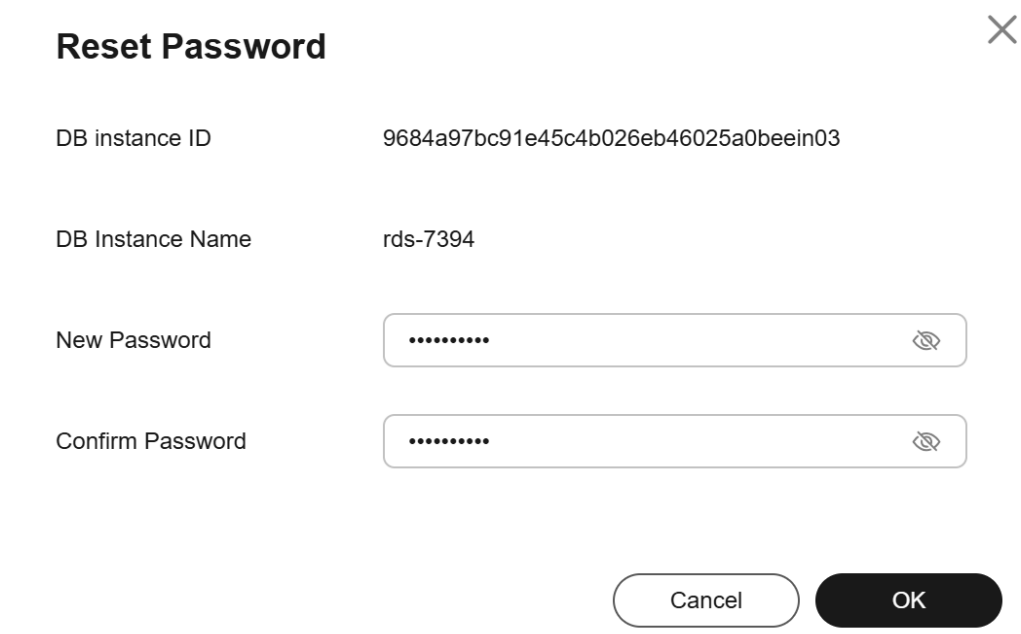
1. Since no password is configured in **Step 1: Buy an RDS for PostgreSQL DB Instance**, you need to reset the password before connecting to the instance. In the instance list, choose **More** > **Reset Password**.

Figure 3-17 Instance list



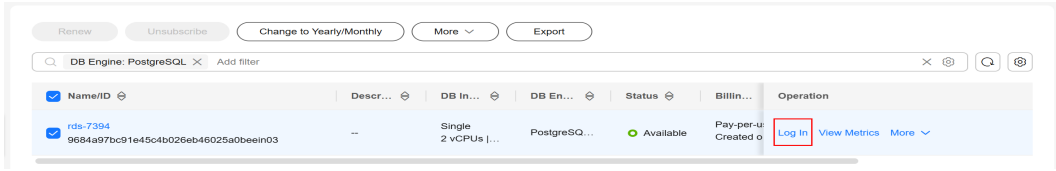
2. Enter a new password, confirm the password, and click **OK**.

Figure 3-18 Resetting a password



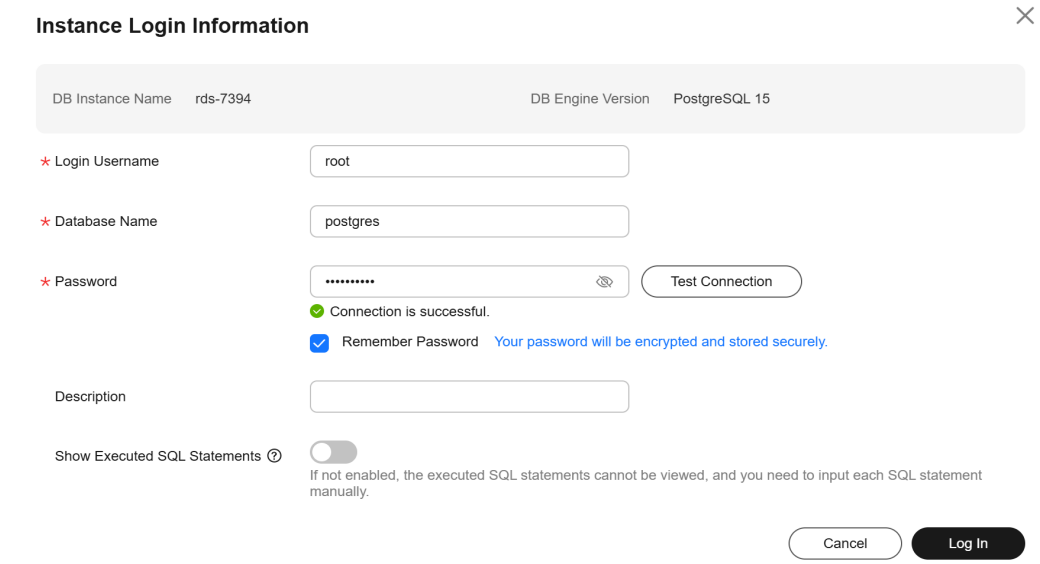
3. Click **Log In** in the **Operation** column.

Figure 3-19 Instance list



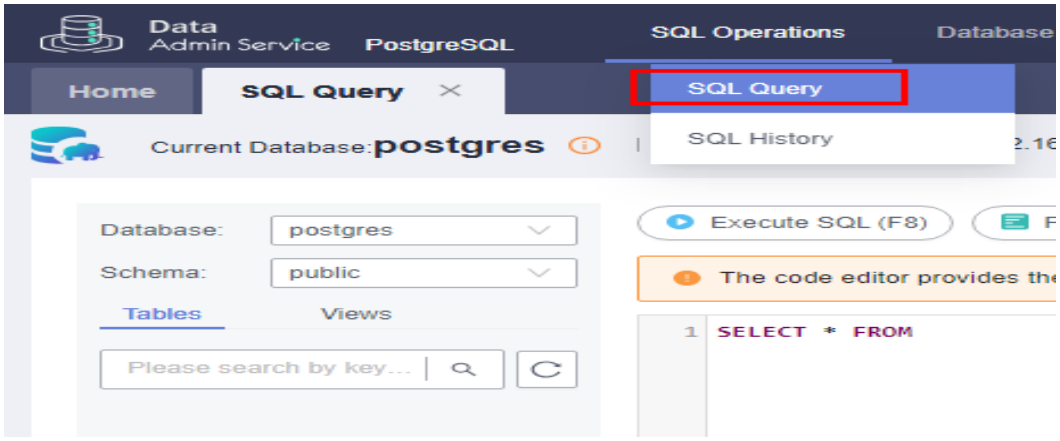
4. Enter the required information and click **Log In**.
   - **Login Username**: Enter **root**.
   - **Database Name**: Enter **postgres**.
   - **Password**: Enter the password you specified in **2**.

Figure 3-20 Logging in to an instance



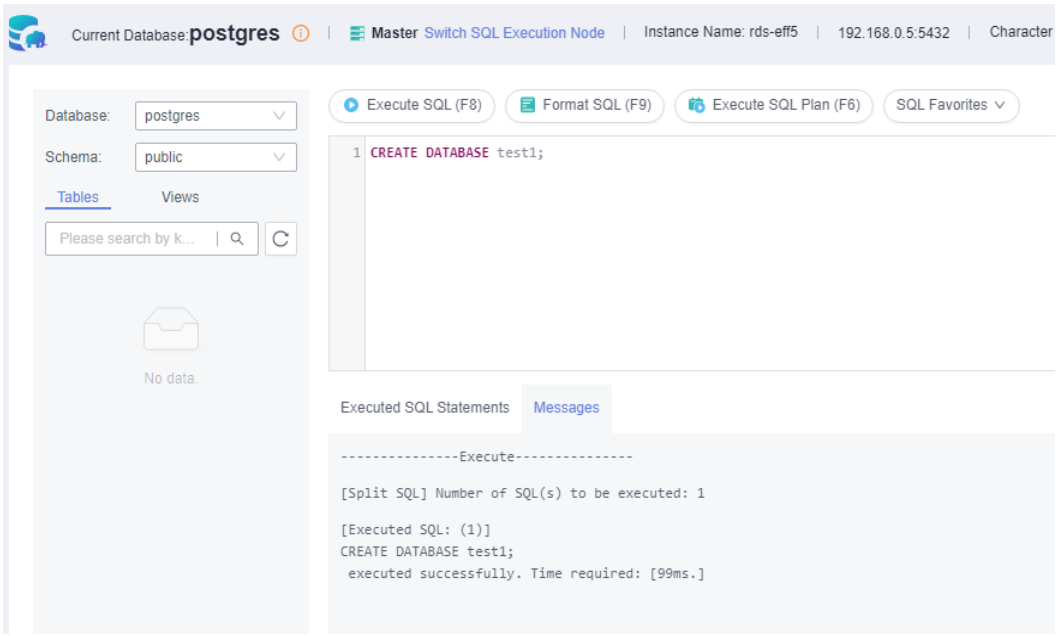
5. Choose **SQL Operations** > **SQL Query**.

Figure 3-21 SQL Query



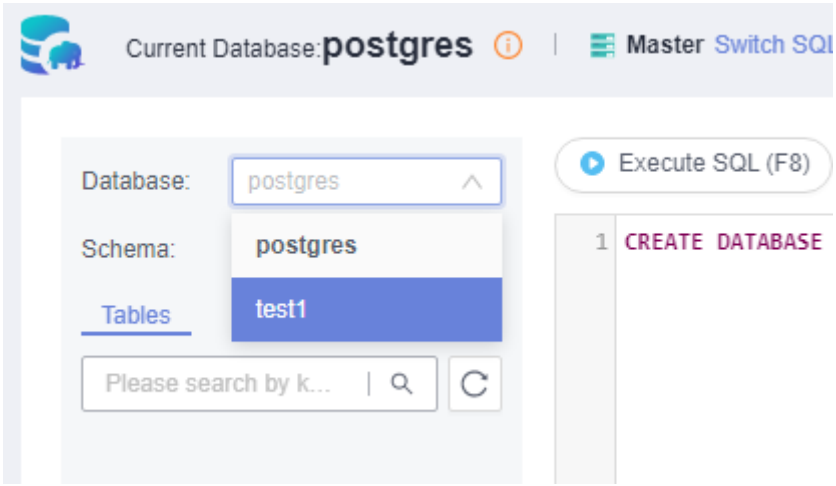
6. Create a database named **test1**.
   CREATE DATABASE test1;

Figure 3-22 Creating a database



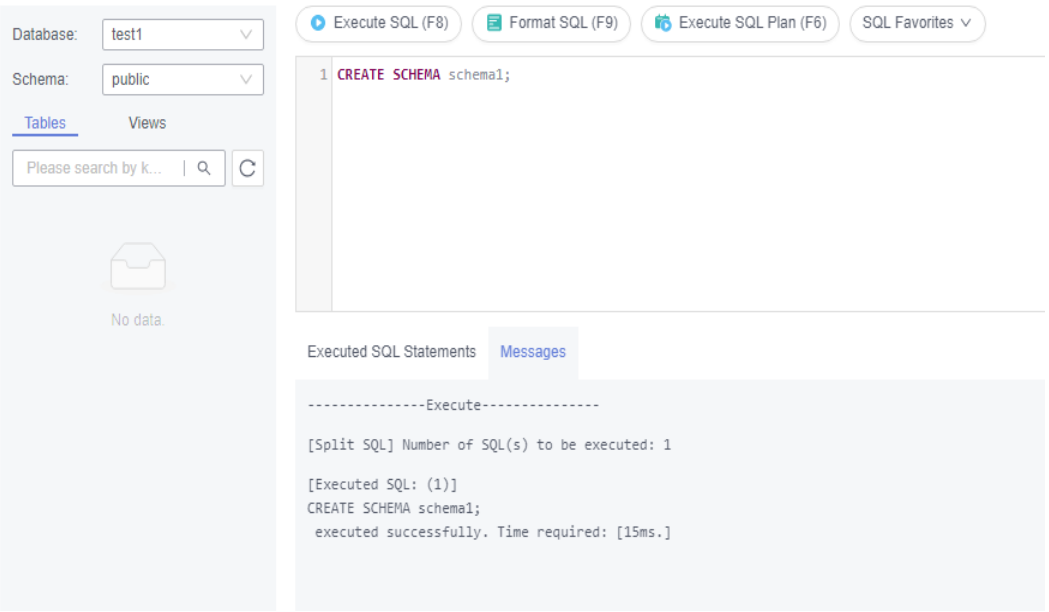
7. Switch to **test1** and create a schema named **schema1** in the database.

Figure 3-23 Switching to the database



CREATE SCHEMA schema1;

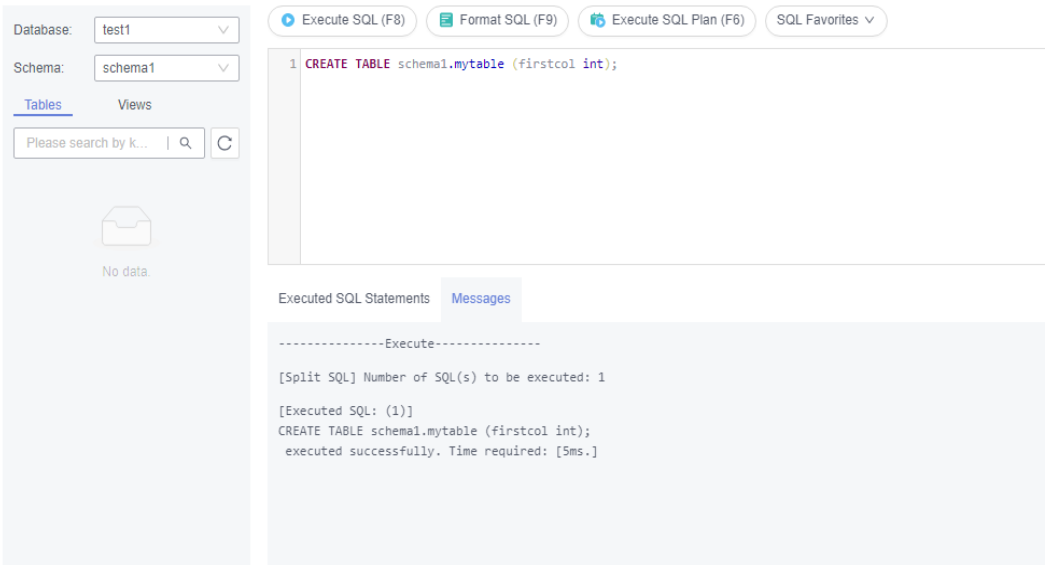
Figure 3-24 Creating a schema



8. Switch to **schema1** and create a table named **mytable** with only one column. Specify the column name as **firstcol** and the column type as **integer**.

   CREATE TABLE schema1.mytable (firstcol int);

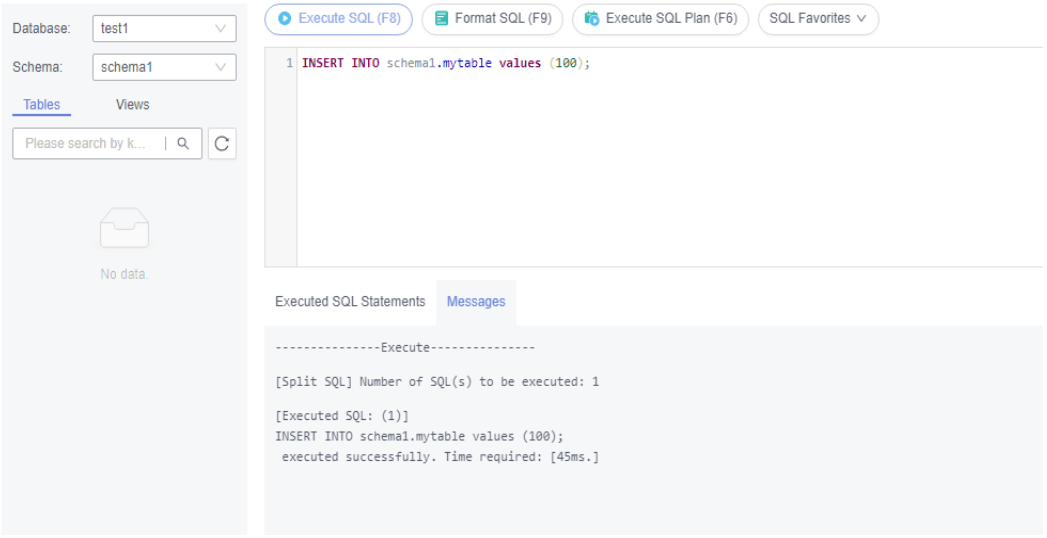
Figure 3-25 Creating a table



9. Insert data to the table.

   INSERT INTO schema1.mytable values (100);

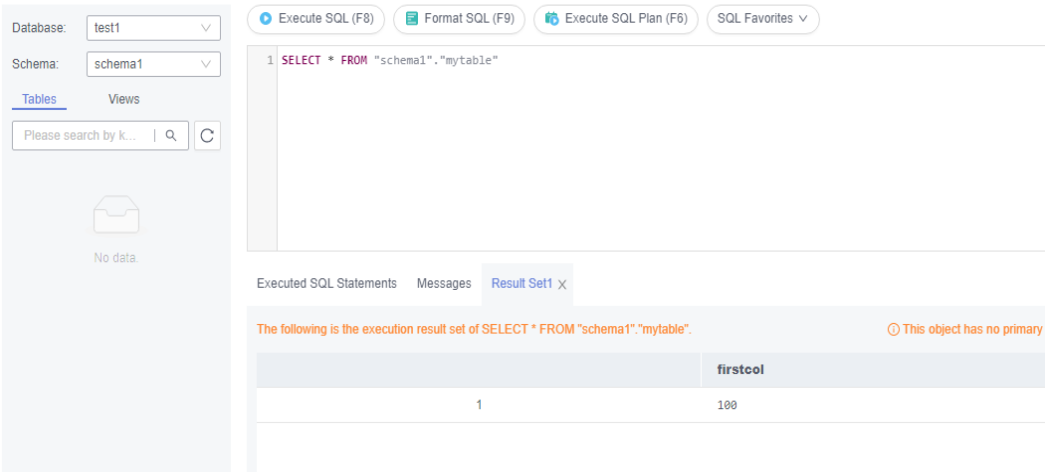
Figure 3-26 Inserting data



Query data in the table.

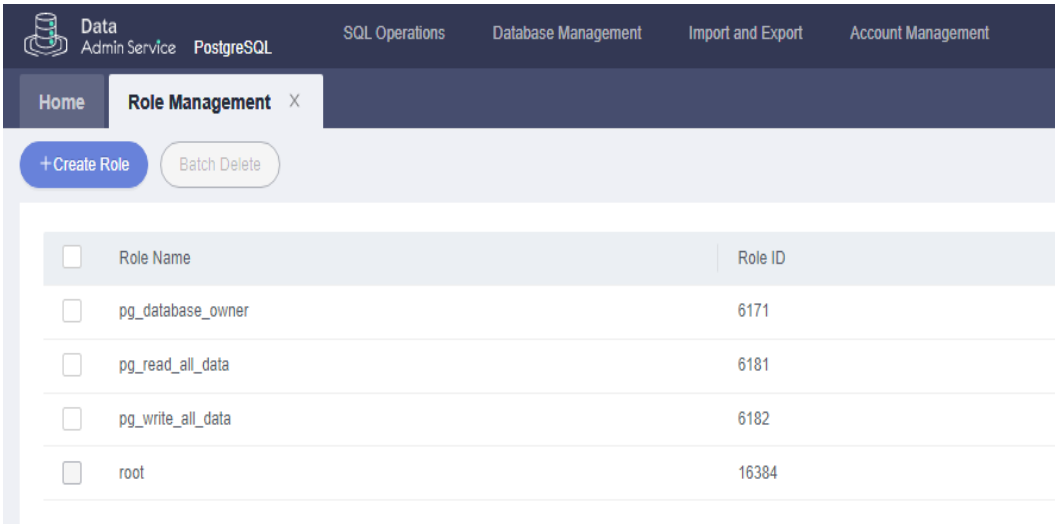
SELECT * FROM "schema1"."mytable"

Figure 3-27 Querying data



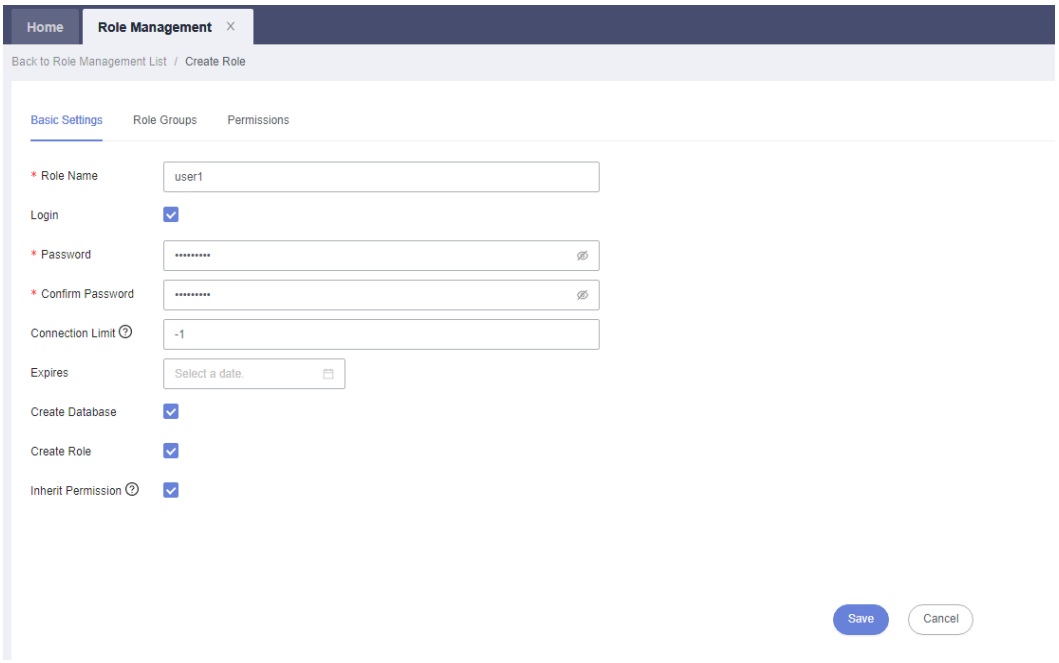
10. In the upper part of the page, choose **Account Management** > **Role Management**.

Figure 3-28 Role management



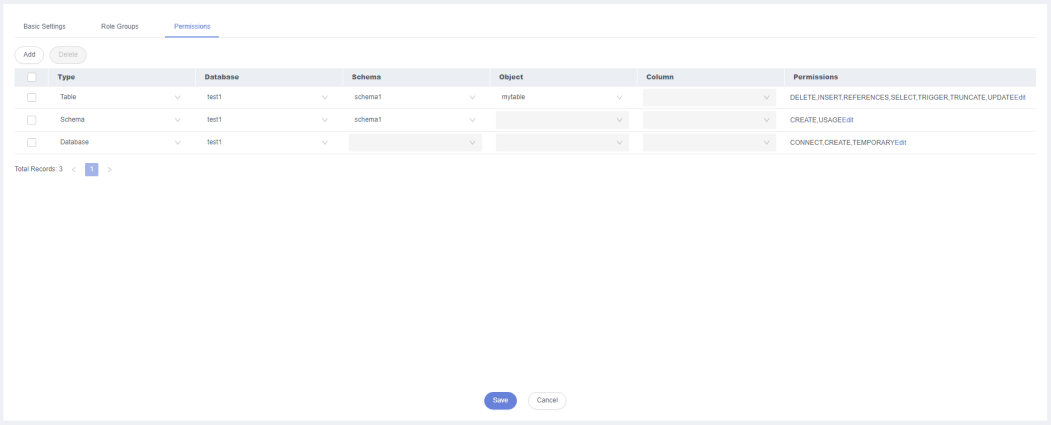
11. Click **Create Role** and complete basic settings. **user1** is used as an example.

Figure 3-29 Creating a role



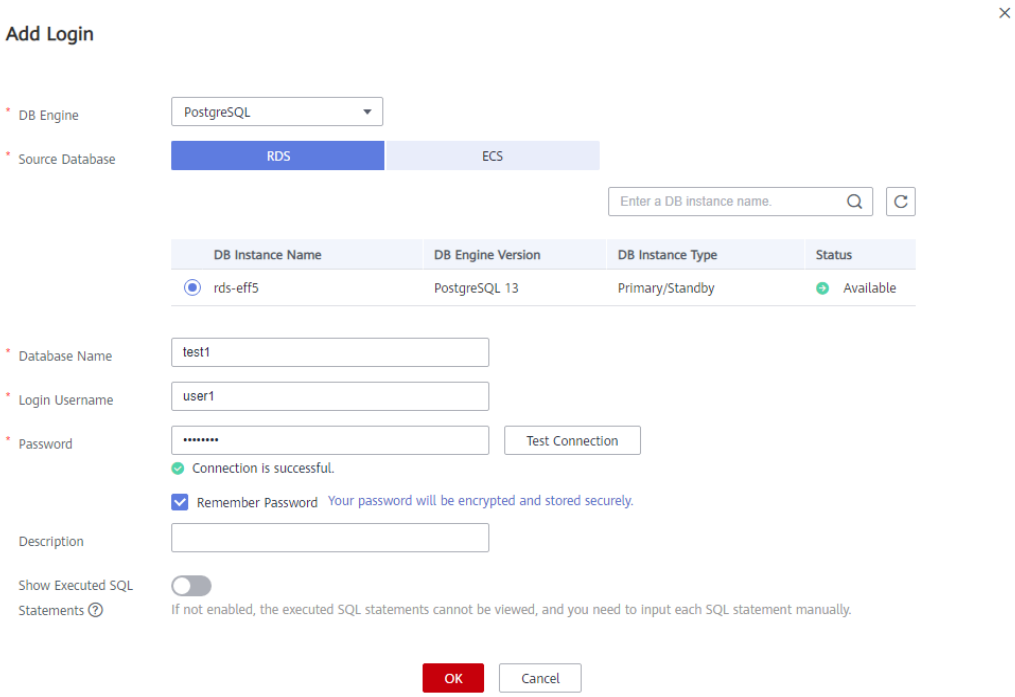
12. Click the **Permissions** tab and grant **user1** the permissions to perform operations on databases, schemas, and tables.

Figure 3-30 Granting permissions



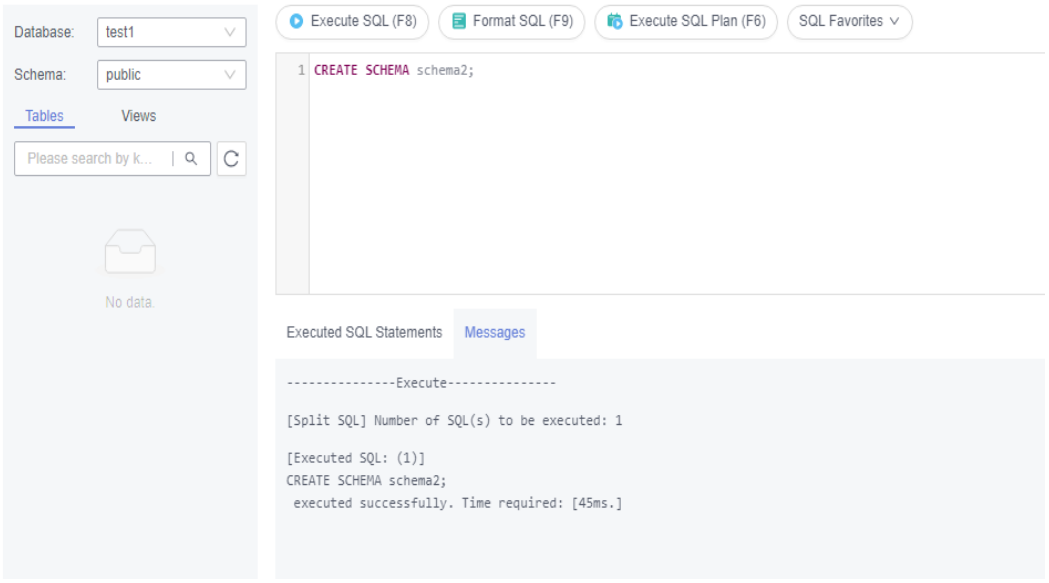
13. On the **Development Tool** page, click **Add Login** and log in to the database as **user1**.

Figure 3-31 Adding login



14. Create **schema2** in **test1** to verify that **user1** has the **CREATE** permission.
CREATE SCHEMA schema2;

Figure 3-32 Verifying permissions



3.3 Getting Started with RDS for PostgreSQL Common Practices

After purchasing and connecting to a DB instance, you can view common practices to better use RDS for PostgreSQL.

Table 3-2 Common practices

Practice		Description
Suggestions on using RDS for PostgreSQL	**Instance Usage Suggestions**	This practice provides suggestions on using RDS for PostgreSQL in terms of database connection, read replicas, reliability and availability, logical replication, database age, stability, routine O&M, and security.
	Database Usage Suggestions	This practice provides suggestions on database naming, table design, index design, SQL design, and security.
Data migration	**Migrating Data to RDS for PostgreSQL Using psql**	This practice describes how to use pg_dump to copy data from the source to an RDS for PostgreSQL DB instance.
	Migrating Data to RDS for PostgreSQL Using the Export and Import Functions of DAS	This practice describes how to use Data Admin Service (DAS) to export data from the source and then import the data to an RDS for PostgreSQL DB instance.

Practice		Description
	From RDS for PostgreSQL to RDS for PostgreSQL	This practice describes how to use DRS to synchronize data from the source to an RDS for PostgreSQL DB instance.
	From Self-Managed PostgreSQL to RDS for PostgreSQL	This practice describes how to use DRS to synchronize data from a self-managed PostgreSQL database to an RDS for PostgreSQL DB instance.
	From PostgreSQL on Other Clouds to RDS for PostgreSQL	This practice describes how to use DRS to synchronize data from PostgreSQL databases on other clouds to RDS for PostgreSQL.
	From Oracle to RDS for PostgreSQL	This practice describes how to use DRS to synchronize data from a self-managed Oracle database to an RDS for PostgreSQL DB instance.
	From RDS for MySQL to RDS for PostgreSQL	This practice describes how to use DRS to synchronize data from an RDS for MySQL DB instance to an RDS for PostgreSQL DB instance.
	From Self-Managed MySQL to RDS for PostgreSQL	This practice describes how to use DRS to synchronize data from a self-managed MySQL database to an RDS for PostgreSQL DB instance.
	From MySQL on Other Clouds to RDS for PostgreSQL	This practice describes how to use DRS to synchronize data from MySQL databases on other clouds to RDS for PostgreSQL.
Data backup	Intra-region automated backup	This practice describes how RDS for PostgreSQL automatically creates backups for a DB instance during a backup window and saves the backups based on the configured retention period.
	Intra-region manual backup	This practice describes how to create manual backups for a DB instance. These backups can be used to restore data for improved reliability.
Data restoration	Restoring from Full Backups to RDS for PostgreSQL Instances	This practice describes how to use an automated or manual backup to restore a DB instance to how it was when the backup was created. The restoration is at the instance level.

Practice		Description
	Restoring a DB Instance to a Point in Time	This practice describes how to use an automated backup to restore instance data to a specified point in time.

4 Getting Started with RDS for SQL Server

4.1 Overview

An RDS DB instance can be connected through a private network, Data Admin Service (DAS), or a public network.

Table 4-1 RDS connection methods

Connect Through	IP Address	Scenarios	Description
DAS	No IP address is required. You can log in to the DAS console and use RDS directly.	DAS enables you to manage databases on a web-based console and provides you with database development, O&M, and intelligent diagnosis to make it easy to use and maintain your databases. The permissions required for connecting to DB instances through DAS are enabled by default.	• Easy to use, secure, advanced, and intelligent • Recommended
Private network	Floating IP	RDS provides a floating IP address by default. When your applications are deployed on an ECS that is in the same region and VPC as RDS, you are advised to use a floating IP address to connect to the RDS DB instance through the ECS.	• Secure and excellent performance • Recommended

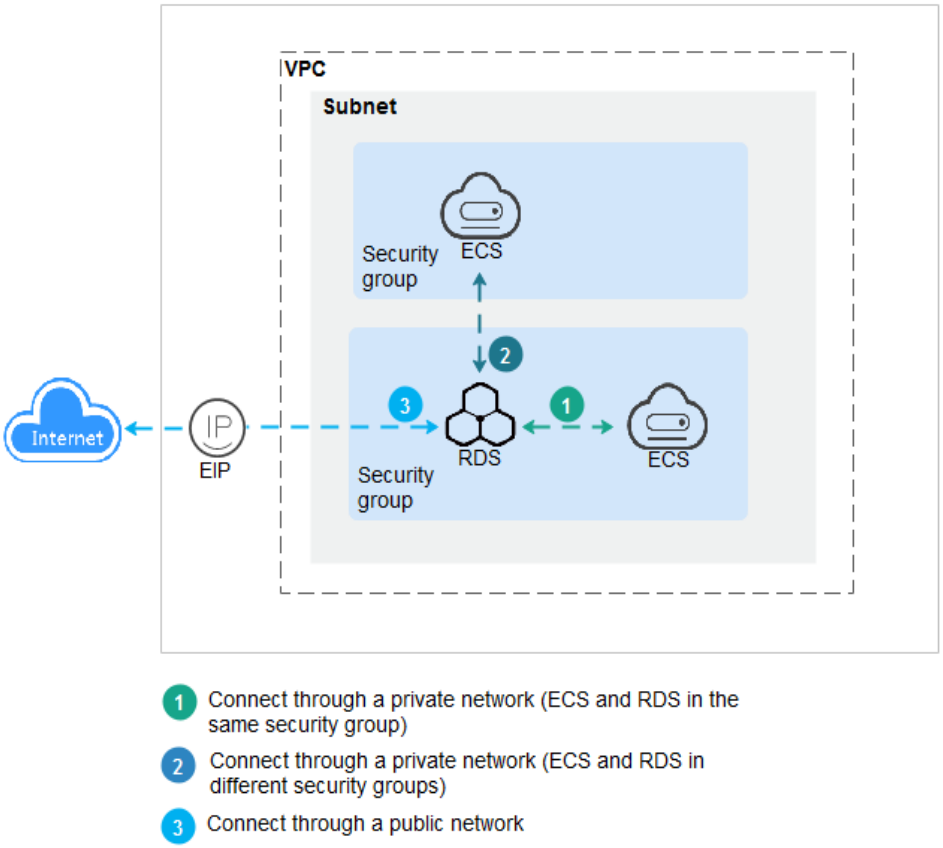
Connect Through	IP Address	Scenarios	Description
Public network	EIP	If you cannot access an RDS DB instance through a floating IP address, bind an EIP to the DB instance and connect the DB instance to the ECS through the EIP.	• A relatively lower level of security compared to other connection methods • To achieve a higher transmission rate and security level, you are advised to migrate your applications to an ECS that is in the same VPC as your RDS DB instance and use a floating IP address to access the DB instance. • You need to purchase an EIP. For details, see **EIP billing details**.

NOTE

- VPC: indicates the Virtual Private Cloud.
- ECS: indicates the Elastic Cloud Server.
- You can log in to DB instances using the Data Admin Service (DAS) service or other database clients.
- If the ECS is in the same VPC as the RDS DB instance, you do not need to apply for an EIP.

Figure 4-1 illustrates the connection over a private network or a public network.

Figure 4-1 DB instance connection



4.2 Connecting to a DB Instance Through DAS (Recommended)

Scenarios

Data Admin Service (DAS) enables you to connect to and manage databases with ease on a web-based console. The permissions required for connecting to DB instances through DAS are enabled by default. You are advised to use this connection method.

Procedure

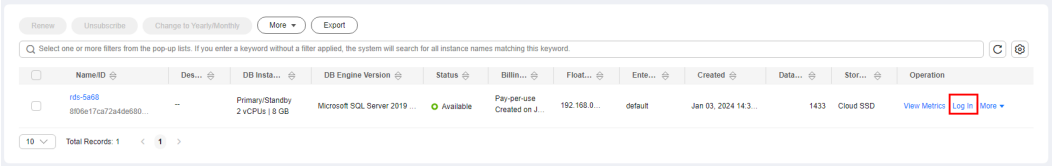
Step 1 **Log in to the management console**.

Step 2 Click in the upper left corner and select a region.

Step 3 Click in the upper left corner of the page and choose **Databases** > **Relational Database Service**.

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

Figure 4-2 Logging in to an instance



Alternatively, click the DB instance name on the **Instances** page. On the displayed **Basic Information** page, click **Log In** in the upper right corner.

Figure 4-3 Logging in to an instance

Step 5 On the displayed login page, enter the correct username and password and click **Log In**.

----End

FAQ

- **What Should I Do If I Can't Connect to My DB Instance Due to Insufficient Permissions?**
- **What Should I Do If I Can't Connect to My RDS for SQL Server Instance?**

Follow-up Operations

After logging in to the DB instance, you can create or migrate databases.

- **Managing RDS for SQL Server Databases Using DAS**
- **Migration Solution Overview**

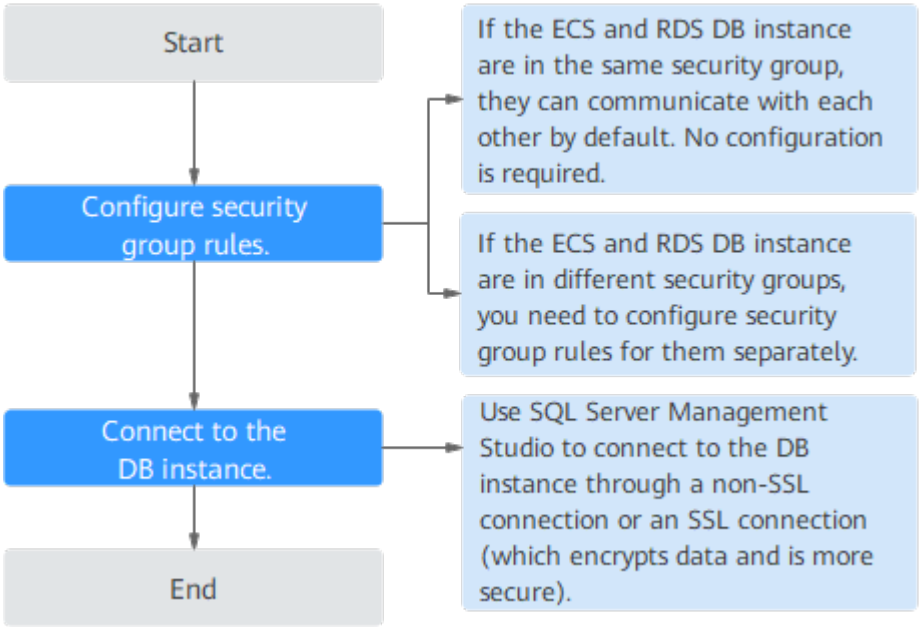
4.3 Connecting to a DB Instance Through a Private Network

4.3.1 Connecting to a DB Instance Through a Private Network

Process

Figure 4-4 illustrates the process of connecting to an RDS for SQL Server DB instance through a private network.

Figure 4-4 Connecting to a DB instance through a private network



4.3.2 Connecting to a DB Instance from a Windows ECS

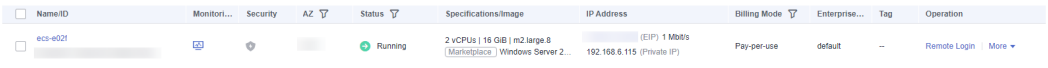
You can connect to your DB instance using a Windows ECS installed with SQL Server Management Studio over a private network.

This section describes how to connect to a DB instance with SSL disabled. To connect to a DB instance with SSL enabled, see **Connecting to an Instance Through a Private Network**.

Step 1: Buy an ECS

1. **Log in to the management console** and check whether there is an ECS available.
   - If there is a Windows ECS, go to **3**.
   - If no Windows ECS is available, go to **2**.

   Figure 4-5 ECS

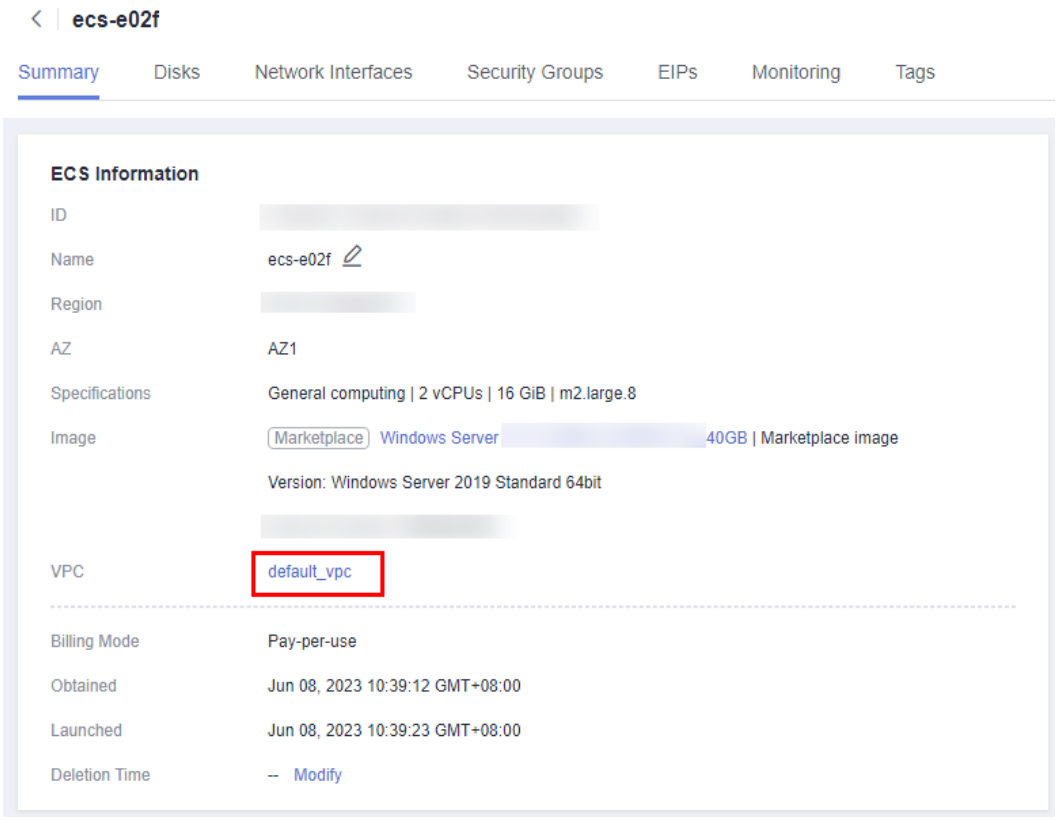
2. Buy an ECS and select Windows as its OS.

   To download SQL Server Management Studio to the ECS, bind an EIP to the ECS. The ECS must be in the same region, VPC, and security group as the RDS for SQL Server DB instance for mutual communications.

   For details about how to purchase a Windows ECS, see **Purchasing a Custom ECS** in *Elastic Cloud Server User Guide*.

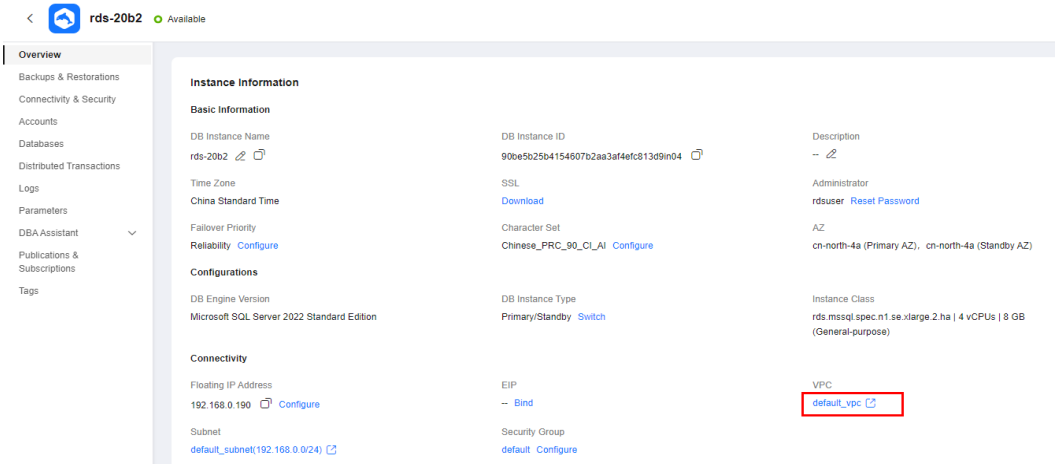
3. On the **ECS Information** page, view the region and VPC of the ECS.

Figure 4-6 ECS information



4. On the **Overview** page of the RDS for SQL Server instance, view the region and VPC of the DB instance.

Figure 4-7 Overview



5. Check whether the ECS and RDS for SQL Server instance are in the same region and VPC.
   - If yes, go to **Step 2: Test Connectivity and Install SQL Server Management Studio**.
   - If they are not in the same region, purchase another ECS or DB instance. The ECS and DB instance in different regions cannot communicate with

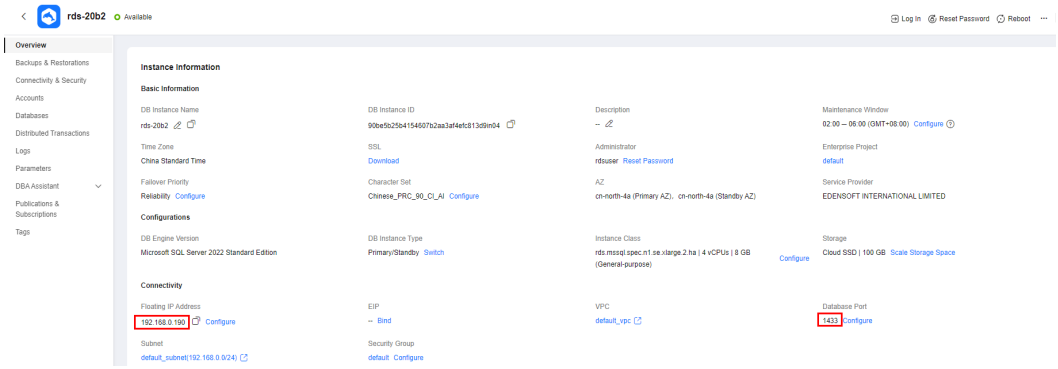
each other. To reduce network latency, deploy your DB instance in the region nearest to your workloads.

- If the ECS and DB instance are in different VPCs, change the VPC of the ECS to that of the DB instance. For details, see **Changing a VPC**.

Step 2: Test Connectivity and Install SQL Server Management Studio

1. Log in to the ECS. For details, see **Login Using VNC** in the *Elastic Cloud Server User Guide*.
2. On the **Instances** page of the RDS console, click the DB instance name to go to the **Overview** page.
3. Obtain the floating IP address and database port of the DB instance.

   Figure 4-8 Connection information

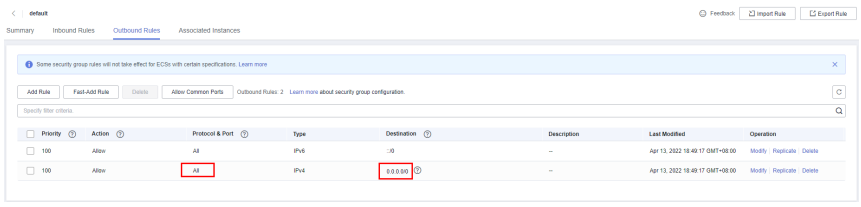
4. Open the cmd window on the ECS and check whether the floating IP address and database port of the DB instance can be connected.

   telnet *192.168.2.182 1433*

   - If yes, network connectivity is normal.
   - If no, check the security group rules.

     - If in the security group of the ECS, there is no outbound rule with **Destination** set to **0.0.0.0/0** and **Protocol & Port** set to **All**, add an outbound rule for the floating IP address and port of the DB instance.

       Figure 4-9 ECS security group

     - If in the security group of the DB instance, there is no inbound rule allowing the access from the private IP address and port of the ECS, add an inbound rule for the private IP address and port of the ECS. For details, see **Configuring Security Group Rules**.

5. Open a browser on the ECS, visit the **Microsoft website**, and download the installation package, for example, SQL Server Management Studio 18.0.

6. Double-click the installation package and complete the installation as instructed.

Step 3: Connect to the DB Instance Using SQL Server Management Studio

1. Start SQL Server Management Studio.
2. Choose **Connect** > **Database Engine**. In the displayed dialog box, enter login information.

Figure 4-10 Connecting to the server

Table 4-2 Parameter description

Parameter	Description
Server name	Floating IP address and database port obtained in **3**.
Authentication	Authentication mode. Select **SQL Server Authentication**.
Login	Name of the account used to access the DB instance. The default value is **rdsuser**.
Password	Password of the account.

3. Click **Connect** to connect to the DB instance.

Follow-up Operations

After logging in to the DB instance, you can create or migrate databases.

- **Managing RDS for SQL Server Databases Using DAS**
- **Migration Solution Overview**

4.3.3 Configuring Security Group Rules

Scenarios

A security group is a collection of access control rules for ECSs and RDS DB instances that have the same security protection requirements and are mutually trusted within a VPC.

This section describes how to create a security group to enable specific IP addresses and ports to access RDS.

First check whether the ECS and RDS DB instance are in the same security group.

- If the ECS and RDS DB instance are in the same security group, they can communicate with each other by default. No security group rule needs to be configured. Go to **Connecting to a DB Instance from a Windows ECS**.
- If the ECS and RDS DB instance are in different security groups, you need to configure security group rules for them, separately.
  - RDS DB instance: Configure an **inbound rule** for the security group with which the RDS DB instance is associated.
  - ECS: The default security group rule allows all outgoing data packets. In this case, you do not need to configure a security group rule for the ECS. If not all outbound traffic is allowed in the security group, you need to configure an **outbound rule** for the ECS.

This section describes how to configure an inbound rule for an RDS DB instance.

For details about the requirements of security group rules, see the **Adding a Security Group Rule** section in the *Virtual Private Cloud User Guide*.

Precautions

The default security group rule allows all outgoing data packets. ECSs and RDS DB instances can access each other if they are deployed in the same security group. After a security group is created, you can configure security group rules to control access from and to the DB instances in the security group.

- By default, you can create a maximum of 100 security groups in your cloud account.
- By default, you can add up to 50 security group rules to a security group.
- One RDS DB instance can be associated with multiple security groups, and one security group can be associated with multiple RDS DB instances.
- Too many security group rules will increase the first packet latency. You are advised to create no more than 50 rules for each security group.
- To enable access to an RDS DB instance from resources outside the security group, you need to configure an **inbound rule** for the security group associated with the RDS DB instance.

> **NOTE**
>
> To ensure the security of your data and DB instances, you are advised to use the principle of least privilege for database access. Change the default database port **1433**, and set the IP address to the remote server's address or the remote server's smallest subnet address to control the access from the remote server.
>
> The default value of **Source** is **0.0.0.0/0**, indicating that RDS DB instances in the security group can be accessed from any IP address.

For details about the requirements of security group rules, see the **Adding a Security Group Rule** section in the *Virtual Private Cloud User Guide*.

Procedure

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

Step 2 Click in the upper left corner and select a region and a project.

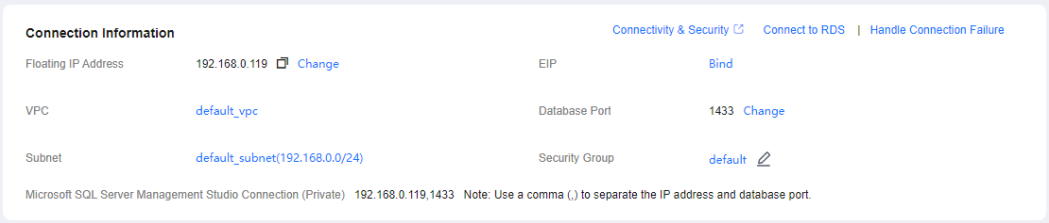
Step 3 Click in the upper left corner of the page and choose **Databases** > **Relational Database Service**.

Step 4 On the **Instances** page, click the DB instance name to go to the **Overview** page.

Step 5 Configure security group rules.

Under **Security Group**, click the security group name.

Figure 4-11 Connection information



Step 6 On the **Inbound Rules** tab, click **Add Rule**. In the displayed dialog box, set required parameters to add an inbound rule.

You can click **+** to add more inbound rules.

Figure 4-12 Adding an inbound rule

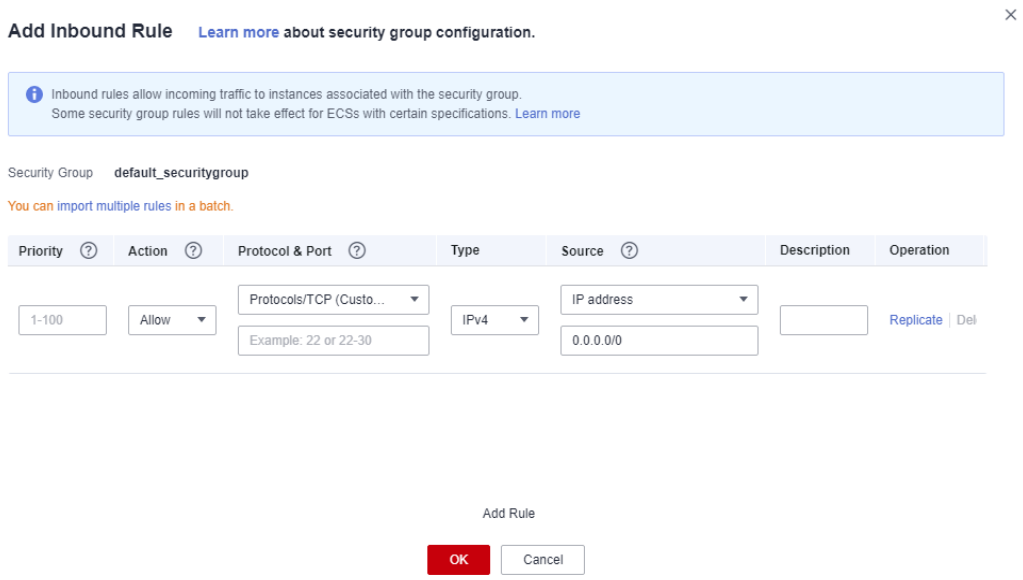


Table 4-3 Inbound rule parameter description

Parameter	Description	Example Value
Priority	Security group rule priority. Value range: 1 to 100. The default priority is 1 and has the highest priority. The security group rule with a smaller value has a higher priority.	1
Action	Security group rule actions. A rule with a deny action overrides another with an allow action if the two rules have the same priority.	Allow
Protocol & Port	**Protocol**: network protocol. Available options: **All ports**, **Custom TCP**, **Custom UDP**, **ICMP**, and **GRE**.	TCP

Parameter	Description	Example Value
	Port: the port over which the traffic can reach your DB instance. An RDS for SQL Server instance can use the default database port 1433 or any port from the range 2100-9500 (excluding 5355 and 5985). If your instance uses 2019 Enterprise Edition, 2019 Standard Edition, 2019 Web Edition, 2017 Enterprise Edition, 2017 Standard Edition, or 2017 Web Edition, ports 5050, 5353, and 5986 cannot be specified for it.	1433
Type	IP address type.	IPv4
Source	Source address. It can be a single IP address, an IP address group, or a security group to allow access from them to your DB instance. Examples: ● Single IP address: 192.168.10.10/32 (IPv4 address) ● IP address segment: 192.168.1.0/24 (IPv4 address segment) ● All IP addresses: 0.0.0.0/0 (any IPv4 address) ● Security group: sg-abc ● IP address group: ipGroup-test	0.0.0.0/0
Description	Supplementary information about the security group rule. This parameter is optional. The description can contain a maximum of 255 characters and cannot contain angle brackets (<) or (>).	-

----End

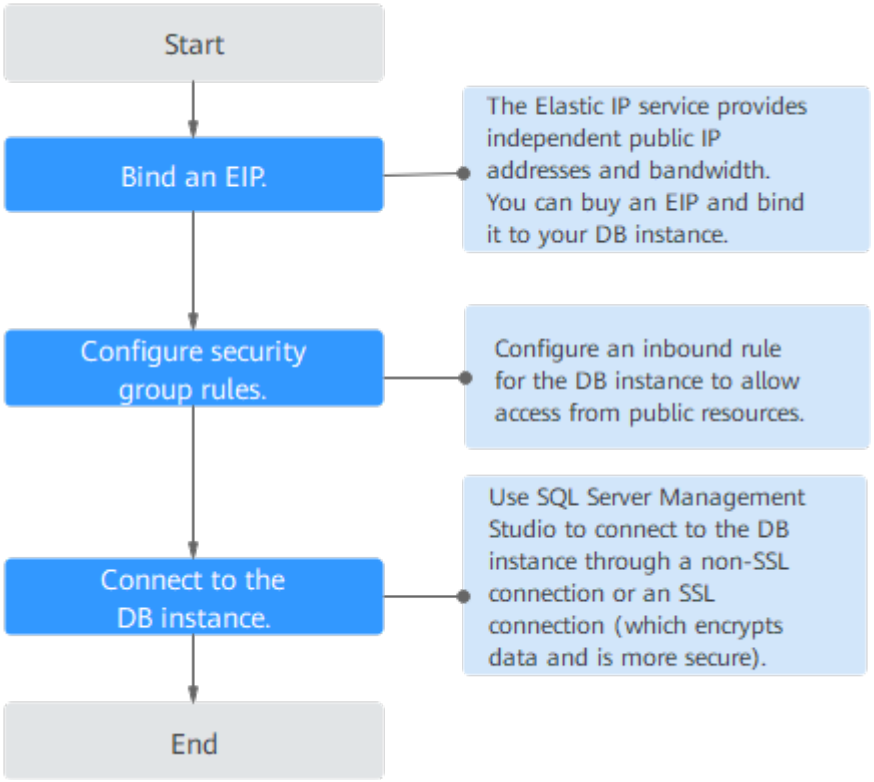
4.4 Connecting to a DB Instance Through a Public Network

4.4.1 Connecting to a DB Instance Through a Public Network

Process

Figure 4-13 illustrates the process of connecting to an RDS for SQL Server DB instance through a public network.

Figure 4-13 Connecting to a DB instance through a public network



4.4.2 Binding an EIP

Scenarios

You can bind an EIP to a DB instance for public accessibility and can unbind the EIP from the DB instance as required.

Precautions

- You need to configure security groups and enable specific IP addresses and ports to access the target DB instance. Before accessing the DB instance, you need to add an individual IP address or an IP address range that will access

the DB instance to the inbound rule. For details, see section **Configuring Security Group Rules**.

- Traffic generated by the public network is charged. You can unbind the EIP from your DB instance when the EIP is no longer used.

Binding an EIP

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

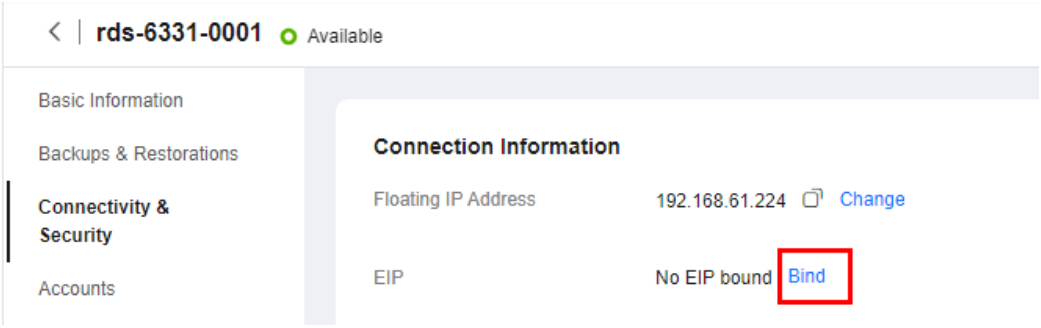
Step 2 Click in the upper left corner and select a region and a project.

Step 3 Click in the upper left corner of the page and choose **Databases** > **Relational Database Service**.

Step 4 On the **Instances** page, click the DB instance name.

Step 5 In the navigation pane on the left, choose **Connectivity & Security**. In the **Connection Information** area, click **Bind** next to the **EIP** field.

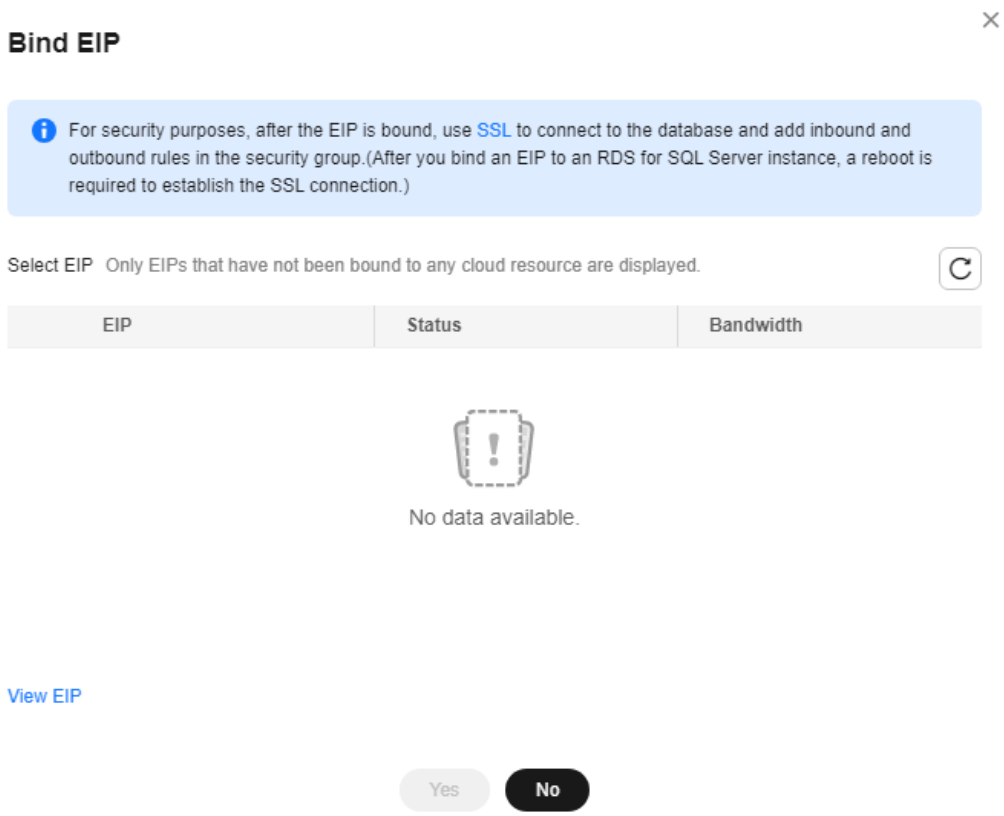
Figure 4-14 Binding an EIP



Step 6 In the displayed dialog box, select an EIP and click **OK**.

If no available EIPs are displayed, click **View EIP** and obtain an EIP.

Figure 4-15 Selecting an EIP



Step 7 On the **Connectivity & Security** page, view the EIP that has been bound to the DB instance.

You can also view the progress and result of binding an EIP to a DB instance on the **Task Center** page.

----End

4.4.3 Connecting to a DB Instance from a Windows Server

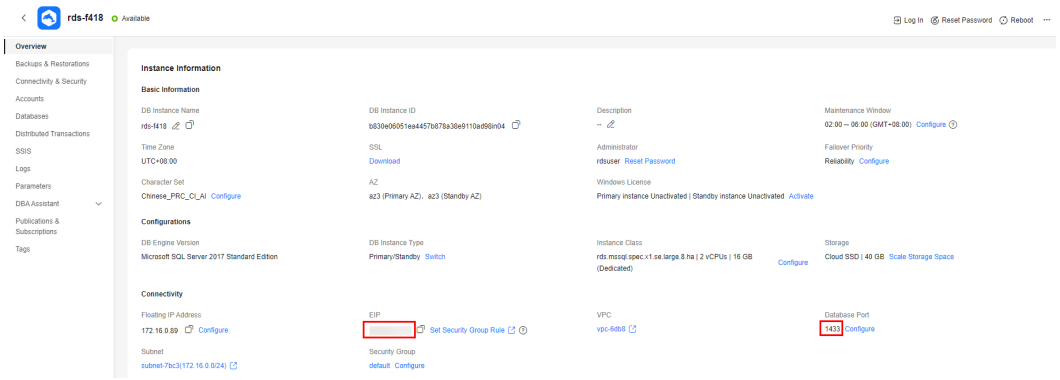
You can connect to your DB instance from a local Windows server installed with SQL Server Management Studio over a public network.

This section describes how to connect to a DB instance with SSL disabled. To connect to a DB instance with SSL enabled, see **Connecting to an Instance Through a Public Network**.

Step 1: Test Connectivity and Install SQL Server Management Studio

1. On the **Instances** page of the RDS console, click the DB instance name to go to the **Overview** page.
2. Obtain the EIP and database port of the DB instance.

Figure 4-16 Connection information



If no EIP has been bound to the DB instance, see **Binding an EIP**.

3. Open the cmd window on your local server and check whether the EIP and database port of the DB instance can be connected.

   telnet *EIP 1433*

   - If yes, network connectivity is normal.
   - If no, check the security group rules.

     Check inbound rules in the security group of the DB instance. Add an inbound rule for the EIP and port of the DB instance. For details, see **Configuring Security Group Rules**.

4. Open a browser on the local server, visit the **Microsoft website**, and download the installation package, for example, SQL Server Management Studio 18.0.
5. Double-click the installation package and complete the installation as instructed.

Step 2: Connect to the DB Instance Using SQL Server Management Studio

1. Start SQL Server Management Studio.
2. Choose **Connect** > **Database Engine**. In the displayed dialog box, enter login information.

Figure 4-17 Connecting to the server

Table 4-4 Parameter description

Parameter	Description
Server name	EIP and database port obtained in 2.
Authentication	Authentication mode. Select **SQL Server Authentication**.
Login	Name of the account used to access the DB instance. The default value is **rdsuser**.
Password	Password of the account.

3. Click **Connect** to connect to the DB instance.

Follow-up Operations

After logging in to the DB instance, you can create or migrate databases.

- **Managing RDS for SQL Server Databases Using DAS**
- **Migration Solution Overview**

4.4.4 Configuring Security Group Rules

Scenarios

A security group is a collection of access control rules for ECSs and RDS DB instances that have the same security protection requirements and are mutually trusted within a VPC.

This section describes how to create a security group to enable specific IP addresses and ports to access RDS.

When you attempt to connect to an RDS DB instance through an EIP, you need to configure an **inbound rule** for the security group associated with the DB instance.

Precautions

The default security group rule allows all outgoing data packets. ECSs and RDS DB instances can access each other if they are deployed in the same security group. After a security group is created, you can configure security group rules to control access from and to the DB instances in the security group.

- By default, you can create a maximum of 100 security groups in your cloud account.
- By default, you can add up to 50 security group rules to a security group.
- One RDS DB instance can be associated with multiple security groups, and one security group can be associated with multiple RDS DB instances.
- Too many security group rules will increase the first packet latency. You are advised to create no more than 50 rules for a security group.
- To enable access to an RDS DB instance from resources outside the security group, you need to configure an **inbound rule** for the security group associated with the RDS DB instance.

NOTE

To ensure the security of your data and DB instances, you are advised to use the principle of least privilege for database access. Change the default database port **1433**, and set the IP address to the remote server's address or the remote server's smallest subnet address to control the access from the remote server.

The default value of **Source** is **0.0.0.0/0**, indicating that RDS DB instances in the security group can be accessed from any IP address.

For details about the requirements of security group rules, see the **Adding a Security Group Rule** section in the *Virtual Private Cloud User Guide*.

Procedure

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

Step 2 Click in the upper left corner and select a region and a project.

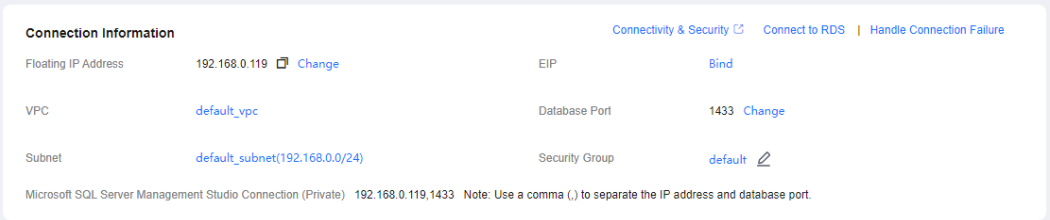
Step 3 Click in the upper left corner of the page and choose **Databases** > **Relational Database Service**.

Step 4 On the **Instances** page, click the DB instance name to go to the **Overview** page.

Step 5 Configure security group rules.

Under **Security Group**, click the security group name.

Figure 4-18 Connection information



Step 6 On the **Inbound Rules** tab, click **Add Rule**. In the displayed dialog box, set required parameters to add an inbound rule.

You can click **+** to add more inbound rules.

Figure 4-19 Adding an inbound rule

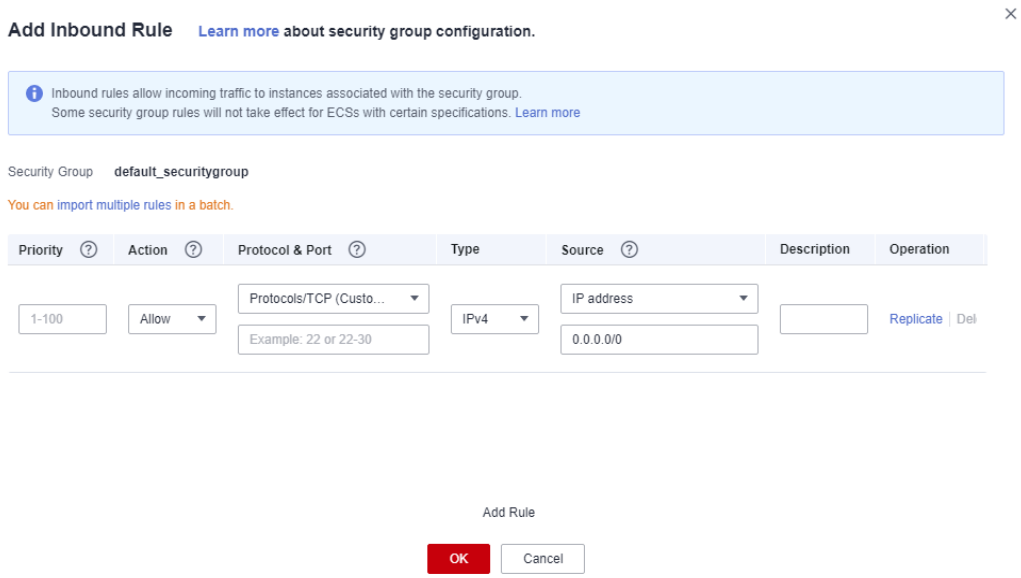


Table 4-5 Inbound rule parameter description

Parameter	Description	Example Value
Priority	Security group rule priority. Value range: 1 to 100. The default priority is 1 and has the highest priority. The security group rule with a smaller value has a higher priority.	1
Action	Security group rule actions. A rule with a deny action overrides another with an allow action if the two rules have the same priority.	Allow
Protocol & Port	**Protocol**: network protocol. Available options: **All ports**, **Custom TCP**, **Custom UDP**, **ICMP**, and **GRE**.	TCP

Parameter	Description	Example Value
	Port: the port over which the traffic can reach your DB instance. An RDS for SQL Server instance can use the default database port 1433 or any port from the range 2100-9500 (excluding 5355 and 5985). If your instance uses 2019 Enterprise Edition, 2019 Standard Edition, 2019 Web Edition, 2017 Enterprise Edition, 2017 Standard Edition, or 2017 Web Edition, ports 5050, 5353, and 5986 cannot be specified for it.	1433
Type	IP address type.	IPv4
Source	Source address. It can be a single IP address, an IP address group, or a security group to allow access from them to your DB instance. Examples: ● Single IP address: 192.168.10.10/32 (IPv4 address) ● IP address segment: 192.168.1.0/24 (IPv4 address segment) ● All IP addresses: 0.0.0.0/0 (any IPv4 address) ● Security group: sg-abc ● IP address group: ipGroup-test	0.0.0.0/0
Description	Supplementary information about the security group rule. This parameter is optional. The description can contain a maximum of 255 characters and cannot contain angle brackets (<) or (>).	-

----End

4.5 Getting Started with RDS for SQL Server Common Practices

After purchasing and connecting to a DB instance, you can view common practices to better use RDS for SQL Server.

Table 4-6 Common practices

Practice		Description
Suggestions on using RDS for SQL Server	**Instance Usage Suggestions**	This practice provides suggestions on DB instance class, database connection, database migration, and instance usage.
SSRS deployment	**Deploying SQL Server Reporting Services (SSRS) on RDS for SQL Server**	This practice describes how to deploy SSRS on RDS for SQL Server.
Data migration	**Migrating Data to RDS for SQL Server Using the Export and Import Functions of DAS**	This practice describes how to use Data Admin Service (DAS) to export data from the source and then import the data to an RDS for SQL Server DB instance.
	Migrating Data from an ECS-Hosted SQL Server Database to RDS for SQL Server Using the Export and Import Functions of SSMS	This practice describes how to use SQL Server Management Studio (SSMS) to migrate data from an ECS-hosted SQL Server database to an RDS for SQL Server DB instance.
	Migrating Data from an On-Premises SQL Server Database to RDS for SQL Server Using the Export and Import Functions of SSMS	This practice describes how to use SSMS to migrate data from an on-premises SQL Server database to an RDS for SQL Server DB instance.
	Deploying SQL Server Reporting Services (SSRS) on RDS for SQL Server	This practice describes how to deploy SSRS on RDS for SQL Server.
	Migrating Backup Data of an RDS for SQL Server DB Instance to Another RDS for SQL Server DB Instance	This practice describes how to use DRS to migrate backup data from the source to an RDS for SQL Server DB instance.

Practice		Description
	From RDS for SQL Server to RDS for SQL Server	This practice describes how to use DRS to synchronize data from an RDS for SQL Server DB instance to another RDS for SQL Server DB instance.
	Migrating Backup Data of an On-Premises SQL Server Database to an RDS for SQL Server DB Instance	This practice describes how to use DRS to migrate backup data of an on-premises SQL Server database to an RDS for SQL Server DB instance.
	From On-Premises SQL Server to RDS for SQL Server	This practice describes how to use DRS to synchronize data from an on-premises SQL Server database to an RDS for SQL Server DB instance.
	Migrating Backup Data of SQL Server Databases on Other Clouds to RDS for SQL Server	This practice describes how to use DRS to migrate backup data of SQL Server databases on other clouds to RDS for SQL Server.
	From SQL Server on Other Clouds to RDS for SQL Server	This practice describes how to use DRS to synchronize data from SQL Server databases on other clouds to RDS for SQL Server.
Data backup	Intra-region automated backup	This practice describes how RDS for SQL Server automatically creates backups for a DB instance during a backup window and saves the backups based on the configured retention period.
	Intra-region manual backup	This practice describes how to create manual backups for a DB instance. These backups can be used to restore data for improved reliability.
Data restoration	Restoring from Full Backups to RDS for SQL Server Instances	This practice describes how to use an automated or manual backup to restore a DB instance to how it was when the backup was created. The restoration is at the instance level.
	Restoring a DB Instance to a Point in Time	This practice describes how to use an automated backup to restore instance data to a specified point in time.

