

Relational Database Service

User Guide

Issue 01
Date 2023-06-19



Copyright © Huawei Cloud Computing Technologies Co., Ltd. 2024. All rights reserved.

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

Trademarks and Permissions



HUAWEI and other Huawei trademarks are 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 Introduction.....	1
1.1 What Is RDS?.....	1
1.2 Basic Concepts.....	2
1.3 Advantages.....	3
1.3.1 Easy Management.....	3
1.3.2 High Security.....	4
1.3.3 High Reliability.....	5
1.3.4 Comparison Between RDS and Self-Built Databases.....	5
1.4 Product Series.....	6
1.4.1 DB Instance Introduction.....	6
1.4.2 Function Comparison.....	7
1.5 DB Instance Description.....	9
1.5.1 DB Instance Types.....	9
1.5.2 DB Instance Classes.....	10
1.5.3 DB Engines and Versions.....	11
1.5.4 DB Instance Statuses.....	12
1.6 Typical Applications.....	13
1.6.1 Read/Write Splitting.....	13
1.7 Constraints.....	13
1.7.1 MySQL Constraints.....	13
1.7.2 PostgreSQL Constraints.....	14
1.8 Related Services.....	15
2 Getting Started with RDS for MySQL.....	16
2.1 Connecting to a DB Instance.....	16
2.2 Connecting to a MySQL DB Instance Through a Private Network.....	18
2.2.1 Overview.....	18
2.2.2 Step 1: Create a DB Instance.....	19
2.2.3 Step 2: Configure Security Group Rules.....	25
2.2.4 Step 3: Connect to a DB Instance Through a Private Network.....	26
2.3 Connecting to a MySQL DB Instance Through a Public Network.....	31
2.3.1 Overview.....	31
2.3.2 Step 1: Create a DB Instance.....	32
2.3.3 Step 2: Bind an EIP.....	37

2.3.4 Step 3: Configure Security Group Rules.....	38
2.3.5 Step 4: Connect to a DB Instance Through a Public Network.....	39
3 Getting Started with RDS for PostgreSQL.....	44
3.1 Connecting to a DB Instance.....	44
3.2 Connecting to a PostgreSQL DB Instance Through a Private Network.....	46
3.2.1 Overview.....	46
3.2.2 Step 1: Create a DB Instance.....	47
3.2.3 Step 2: Configure Security Group Rules.....	52
3.2.4 Step 3: Connect to a DB Instance Through psql.....	53
3.3 Connecting to a PostgreSQL DB Instance Through a Public Network.....	55
3.3.1 Overview.....	56
3.3.2 Step 1: Create a DB Instance.....	57
3.3.3 Step 2: Bind an EIP.....	61
3.3.4 Step 3: Configure Security Group Rules.....	62
3.3.5 Step 4: Connect to a DB Instance Through psql.....	63
4 Working with RDS for MySQL.....	66
4.1 Instance Management.....	66
4.1.1 Changing a DB Instance Name.....	66
4.1.2 Changing the Failover Priority.....	67
4.1.3 Changing a DB Instance Class.....	67
4.1.4 Scaling Up Storage Space.....	68
4.1.5 Rebooting a DB Instance.....	70
4.1.6 Changing a DB Instance Type from Single to Primary/Standby.....	71
4.1.7 Manually Switching Between Primary and Standby DB Instances.....	72
4.1.8 Selecting Displayed Items.....	73
4.1.9 Exporting DB Instance Information.....	74
4.1.10 Creating a Same DB Instance.....	74
4.1.11 Deleting a DB Instance or Read Replica.....	75
4.2 Read Replicas.....	76
4.2.1 Introducing Read Replicas.....	77
4.2.2 Creating a Read Replica.....	78
4.2.3 Managing a Read Replica.....	80
4.3 Backups and Restorations.....	80
4.3.1 Working with Backups.....	80
4.3.2 Configuring an Automated Backup Policy.....	81
4.3.3 Creating a Manual Backup.....	82
4.3.4 Downloading a Backup File.....	84
4.3.5 Downloading a Binlog Backup File.....	85
4.3.6 Restoring a DB Instance from a Backup.....	86
4.3.7 Restoring a DB Instance to a Point in Time.....	88
4.3.8 Replicating a Backup.....	89
4.3.9 Deleting a Manual Backup.....	90

4.4 Parameter Template Management.....	91
4.4.1 Suggestions on Tuning MySQL Parameters.....	91
4.4.2 Creating a Parameter Template.....	92
4.4.3 Modifying Parameters.....	94
4.4.4 Exporting a Parameter Template.....	96
4.4.5 Comparing Parameter Templates.....	97
4.4.6 Viewing Parameter Change History.....	98
4.4.7 Replicating a Parameter Template.....	99
4.4.8 Resetting a Parameter Template.....	99
4.4.9 Applying a Parameter Template.....	100
4.4.10 Viewing Application Records of a Parameter Template.....	101
4.4.11 Modifying a Parameter Template Description.....	101
4.4.12 Deleting a Parameter Template.....	102
4.5 Connection Management.....	102
4.5.1 Configuring and Changing a Floating IP Address.....	102
4.5.2 Binding and Unbinding an EIP.....	103
4.5.3 Changing the Database Port.....	104
4.5.4 Configuring a Security Group Rule.....	105
4.6 Data Migration.....	107
4.6.1 Migrating Data to RDS for MySQL Using mysqldump.....	107
4.7 Data Security.....	110
4.7.1 Resetting the Administrator Password.....	110
4.7.2 Changing a Security Group.....	111
4.8 Metrics and Alarms.....	112
4.8.1 Configuring Displayed Metrics.....	112
4.8.2 Setting Alarm Rules.....	120
4.8.3 Viewing Monitoring Metrics.....	121
4.9 Log Management.....	122
4.9.1 Viewing Error Logs.....	122
4.9.2 Viewing Slow Query Logs.....	123
4.9.3 Viewing Failover/Switchover Logs.....	123
4.10 Task Center.....	124
4.10.1 Viewing a Task.....	124
4.10.2 Deleting a Task Record.....	125
5 Working with RDS for PostgreSQL.....	126
5.1 Instance Management.....	126
5.1.1 Changing a DB Instance Name.....	126
5.1.2 Changing the Failover Priority.....	127
5.1.3 Changing a DB Instance Class.....	127
5.1.4 Scaling Up Storage Space.....	128
5.1.5 Rebooting a DB Instance.....	130
5.1.6 Changing a DB Instance Type from Single to Primary/Standby.....	131

5.1.7 Manually Switching Between Primary and Standby DB Instances.....	132
5.1.8 Selecting Displayed Items.....	133
5.1.9 Exporting DB Instance Information.....	133
5.1.10 Creating a Same DB Instance.....	134
5.1.11 Deleting a DB Instance or Read Replica.....	135
5.2 Read Replicas.....	136
5.2.1 Introducing Read Replicas.....	136
5.2.2 Creating a Read Replica.....	137
5.2.3 Managing a Read Replica.....	139
5.3 Backups and Restorations.....	140
5.3.1 Working with Backups.....	140
5.3.2 Configuring an Automated Backup Policy.....	140
5.3.3 Creating a Manual Backup.....	141
5.3.4 Restoring a DB Instance to a Point in Time.....	143
5.3.5 Restoring a DB Instance from a Backup.....	145
5.3.6 Downloading a Backup File.....	147
5.3.7 Replicating a Backup.....	149
5.3.8 Deleting a Manual Backup.....	150
5.4 Parameter Template Management.....	150
5.4.1 Creating a Parameter Template.....	150
5.4.2 Modifying Parameters.....	152
5.4.3 Exporting a Parameter Template.....	155
5.4.4 Comparing Parameter Templates.....	156
5.4.5 Viewing Parameter Change History.....	157
5.4.6 Replicating a Parameter Template.....	158
5.4.7 Resetting a Parameter Template.....	158
5.4.8 Applying a Parameter Template.....	159
5.4.9 Viewing Application Records of a Parameter Template.....	160
5.4.10 Modifying a Parameter Template Description.....	160
5.4.11 Deleting a Parameter Template.....	161
5.5 Connection Management.....	161
5.5.1 Configuring and Changing a Floating IP Address.....	161
5.5.2 Binding and Unbinding an EIP.....	162
5.5.3 Changing the Database Port.....	163
5.5.4 Connecting to a DB Instance Through pgAdmin.....	164
5.6 Data Migration.....	166
5.6.1 Migrating Data to RDS for PostgreSQL Using psql.....	166
5.7 Data Security.....	168
5.7.1 Resetting the Administrator Password.....	169
5.7.2 Changing a Security Group.....	170
5.8 Metrics and Alarms.....	171
5.8.1 Configuring Displayed Metrics.....	171

5.8.2 Setting Alarm Rules.....	175
5.8.3 Viewing Monitoring Metrics.....	175
5.9 Log Management.....	176
5.9.1 Viewing Slow Query Logs.....	176
5.10 Task Center.....	177
5.10.1 Viewing a Task.....	177
5.10.2 Deleting a Task Record.....	178
5.11 Plugin Management.....	179
5.11.1 Creating and Deleting a Plugin.....	179
5.11.2 Plugins Supported By RDS for PostgreSQL.....	181
6 FAQs.....	184
6.1 Product Consulting.....	184
6.1.1 What Precautions Should Be Taken When Using RDS?.....	184
6.1.2 What Is the Availability of RDS DB Instances?.....	184
6.1.3 Can I Use a Template to Create DB Instances?.....	184
6.1.4 What Are the Differences Between RDS and Other Database Solutions?.....	185
6.1.5 Will My RDS DB Instances Be Affected by Other Users' DB Instances?.....	186
6.1.6 Does RDS Support Cross-AZ High Availability?.....	186
6.1.7 Can RDS Primary/Standby DB Instances Be Changed to Single DB Instances?.....	186
6.1.8 What Should I Do If Garbled Characters Are Displayed After SQL Query Results Are Exported to an Excel File?.....	186
6.1.9 How Do I Create an AD Domain?.....	186
6.1.10 What Can I Do About Websites Responding Slower After Using RDS?.....	192
6.1.11 How Does a Cloud Database Perform a Primary/Standby Switchover?.....	192
6.1.12 Can Multiple ECSs Connect to the Same RDS DB Instance?.....	193
6.1.13 Can Primary and Standby RDS DB Instances Be Deployed in the Same AZ?.....	194
6.2 Resource and Disk Management.....	194
6.2.1 Which Types of Logs and Files Occupy RDS Storage Space?.....	194
6.2.2 Which Items Occupy the Storage Space of My RDS DB Instances?.....	195
6.2.3 What Overhead Does the Storage Space Have After I Applied for an RDS DB Instance?.....	195
6.2.4 How Much Storage Space Is Required for DDL Operations?.....	195
6.2.5 How Many DB Instances Can Run on RDS?.....	195
6.2.6 How Many Databases Can Run on an RDS DB Instance?.....	195
6.3 Database Connection.....	196
6.3.1 Can an External Server Access the RDS Database?.....	196
6.3.2 How Do I Troubleshoot If the Number of RDS Database Connections Reaches the Upper Limit?....	196
6.3.3 What Is the Maximum Number of Connections to an RDS DB Instance?.....	196
6.3.4 How Can I Create and Connect to an ECS?.....	197
6.3.5 What Should I Do If an ECS Cannot Connect to an RDS DB Instance?.....	198
6.3.6 What Should I Do If a Database Client Problem Causes a Connection Failure?.....	198
6.3.7 What Should I Do If an RDS Database Problem Causes a Connection Failure?.....	199
6.3.8 How Do My Applications Access an RDS DB Instance in a VPC?.....	199

6.3.9 Do Applications Need to Support Reconnecting to the RDS DB Instance Automatically?.....	199
6.3.10 How Can I Connect to a PostgreSQL Database Through JDBC?.....	199
6.3.11 Can I Access an RDS DB Instance Over an Intranet Across Regions?.....	202
6.3.12 Is an SSL Connection to a DB Instance Interrupted After a Primary/Standby Switchover or Failover Occurs?.....	202
6.3.13 Does MySQL Support SSL Connections?.....	202
6.3.14 Why Does the New Password Not Take Effect After I Reset the Administrator Password?.....	203
6.4 Database Migration.....	203
6.4.1 Why Do I Need to Use the mysqldump and pg_dump Tools for Migration?.....	203
6.4.2 What Types of DB Engines Does RDS Support for Importing Data?.....	203
6.5 Database Permission.....	204
6.5.1 Why Does the Root User Not Have the Super Permission?.....	204
6.6 Database Storage.....	204
6.6.1 What Storage Engines Does the RDS for MySQL Support?.....	204
6.6.2 What Is the RDS DB Instance Storage Configuration?.....	205
6.6.3 Can I Change the Storage Type of an RDS DB Instance from Common I/O to Ultra-high I/O?.....	205
6.6.4 What Should I Do If My Data Exceeds the Database Storage Space of an RDS DB Instance?.....	206
6.7 Client Installation.....	206
6.7.1 How Can I Install the MySQL Client?.....	206
6.7.2 How Can I Install the PostgreSQL Client?.....	207
6.8 Backup and Restoration.....	209
6.8.1 How Long Does RDS Store Backup Data?.....	209
6.8.2 Can My Database Be Used in the Backup Window?.....	209
6.8.3 How Can I Back Up RDS Databases to an ECS?.....	209
6.8.4 Why Has My Automated Backup Failed?.....	210
6.8.5 What Happens to Database Backups After an RDS DB Instance Is Deleted?.....	210
6.8.6 Will My Backups Be Deleted If I Delete My Cloud Account?.....	210
6.8.7 Why Is a Table or Data Missing from My Database?.....	210
6.9 Database Monitoring.....	211
6.9.1 Which DB Instance Monitoring Metrics Do I Need to Pay Most Attention To?.....	211
6.10 Capacity Expansion and Specification Change.....	211
6.10.1 Are My RDS DB Instances Available When Scaling?.....	211
6.10.2 Why Does the DB Instance Become Faulty After the Original Database Port Is Changed?.....	212
6.11 Database Parameter Modification.....	212
6.11.1 What Inappropriate Parameter Settings Cause Unavailability of the PostgreSQL Database?.....	212
6.12 Log Management.....	213
6.12.1 How Long Is the Delay of RDS MySQL Slow Query Logs?.....	213
6.12.2 Can I Export Statistics on RDS Slow Query Logs?.....	213
6.13 Network Security.....	213
6.13.1 What Security Protection Policies Does RDS Have?.....	213
6.13.2 How Can I Ensure the Security of RDS DB Instances in a VPC?.....	213
6.13.3 How Can Data Security Be Ensured During Transmission When I Access RDS Through an EIP?.....	213
6.13.4 How Can I Prevent Untrusted Source IP Addresses from Accessing RDS?.....	214

6.13.5 How Can I Import the Root Certificate to the Windows or Linux OS?.....	214
6.13.6 How Can I Identify the Validity Period of an SSL Root Certificate?.....	215
6.13.7 How Can I Identify Data Corruption?.....	215

1 Introduction

1.1 What Is RDS?

RDS is a cloud-based web service that is reliable, scalable, easy to manage, and immediately ready for use. RDS supports the following DB engines:

- [MySQL](#)
- [PostgreSQL](#)

RDS provides a comprehensive performance monitoring system, multi-level security protection measures, and a professional database management platform, allowing you to easily set up and scale a relational database. On the RDS console, you can perform almost all necessary tasks and no programming is required. The console simplifies operation procedures and reduces routine O&M workloads, so that you can focus on your application and service development.

RDS for MySQL

MySQL is one of the world's most popular open-source relational databases. It works with the Linux, Apache, and Perl/PHP/Python (LAMP) model to provide efficient web solutions. RDS for MySQL is reliable, secure, scalable, inexpensive, easy to manage, and immediately ready for use.

- It supports various web applications and is cost-effective, preferred by small- and medium-sized enterprises.
- A web-based console provides comprehensive monitoring information, making your operations easy and visual.
- You can flexibly adjust the required resources based on your service requirements and pay for only what you use.

For details about the versions supported by RDS for MySQL, see section [DB Engines and Versions](#).

RDS for PostgreSQL

PostgreSQL is an open-source object-relational database management system with an emphasis on extensibility and standards compliance. It is known as the

most advanced open-source database. RDS for PostgreSQL applies to business-oriented online transaction processing (OLTP) scenarios and supports NoSQL (JSON, XML, or hstore) and geographic information system (GIS) data types. It has earned a reputation for reliability and data integrity, and applies to Internet websites, location-based applications, and complex data object processing.

- RDS for PostgreSQL supports the postgis plugin and provides excellent spatial performance.
- RDS for PostgreSQL applies to various scenarios and is cost-effective. You can flexibly scale resources based on your service requirements and pay for only what you use.

For details about the versions supported by RDS for PostgreSQL, see section [DB Engines and Versions](#).

1.2 Basic Concepts

DB Instances

DB instances are the smallest management units used by RDS. Each is an isolated database environment on the cloud. You can use RDS to create and manage DB instances running various DB engines. For details about DB instance types, specifications, engines, versions, and statuses, see [DB Instance Description](#).

DB Engines

RDS supports the following DB engines:

- MySQL
- PostgreSQL

For details about the supported versions, see [DB Engines and Versions](#).

DB Instance Types

RDS DB instances are classified into two types: single and primary/standby. Different series support different DB engines and instance specifications.

For details about the product series, see [DB Instance Introduction](#) and [Function Comparison](#).

DB Instance Classes

For details about supported DB engines and instance classes, including the number of vCPUs and memory (GB), see [DB Instance Classes](#).

Automated Backups

When you create a DB instance, an automated backup policy is enabled by default. After the DB instance is created, you can modify the policy. RDS will automatically create full backups for DB instances based on your settings.

Manual Backups

Manual backups are user-initiated full backups of DB instances. They are retained until you delete them manually.

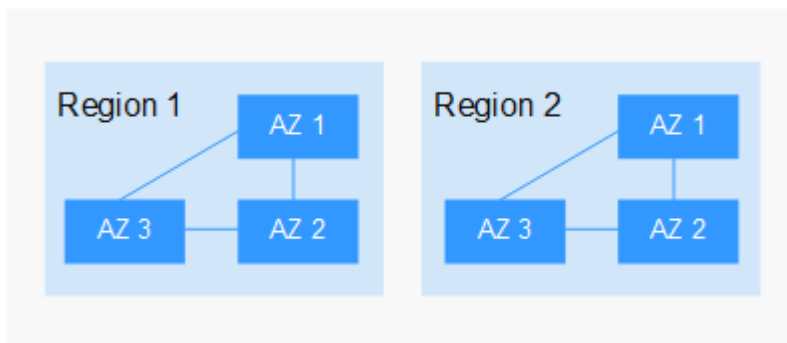
Regions and AZs

A region and availability zone (AZ) identify the location of a data center. You can create resources in a specific region and AZ.

- A region is a physical data center. Each region is completely independent, improving fault tolerance and stability. After a resource is created, its region cannot be changed.
- An AZ is a physical location using independent power supplies and networks. Faults in an AZ do not affect other AZs. A region can contain multiple AZs, which are physically isolated but interconnected through internal networks. This ensures the independence of AZs and provides low-cost and low-latency network connections.

Figure 1-1 shows the relationship between regions and AZs.

Figure 1-1 Regions and AZs



Projects

Projects are used to group and isolate OpenStack resources (computing resources, storage resources, and network resources). A project can be a department or a project team. Multiple projects can be created for one account.

1.3 Advantages

1.3.1 Easy Management

Instant Availability

You can obtain a DB instance on the management console and access RDS through an ECS to reduce the application response time and save the traffic fees for public access.

Elastic Scaling

You can flexibly adjust the required resources based on your service requirements and pay for only what you use. Cloud Eye can monitor the changes of database pressure and data storage space. You can adjust DB instance classes accordingly.

High Compatibility

Operation methods of the database engines (DB engines) provided by RDS are the same as those of the native engines. It is also compatible with existing programs and tools.

Easy O&M

Routine RDS maintenance and management operations, including hardware and software fault handling and database patch updates, are easy to perform. With a web-based console, you can reboot DB instances, reset passwords, modify parameters, view error or slow query logs, and restore data. Additionally, the system helps you monitor DB instances in real time and generates alarms if an error occurs. You can check DB instance information at any time, including CPU usage, IOPS, database connections, and storage space usage.

1.3.2 High Security

Network Isolation

RDS uses Virtual Private Cloud (VPC) and network security groups to isolate and secure your DB instances. VPCs allow you to define the IP address range that can access RDS. You can configure subnets and security groups to control access to DB instances.

Access Control

RDS controls access through the account/IAM user and security groups. When you create an RDS DB instance, a domain is automatically created. To separate permissions, you can create IAM users and assign permissions to them as needed. VPC security groups have rules that govern both inbound and outbound traffic of DB instances.

Transmission Encryption

RDS uses Transport Layer Security (TLS) and Secure Sockets Layer (SSL) to offer encryption during transmission. You can download the Certificate Agency (CA) certificate from the RDS console and upload it when connecting to a database for authentication.

Storage Encryption

RDS uses static encryption and tablespace encryption to encrypt the data to be stored. Encryption keys are managed by Key Management Service (KMS).

Data Deletion

When you delete an RDS DB instance, its attached disks, object storage space its backups occupy, and all data it stores will be deleted. The deleted data cannot be viewed or restored.

Anti-DDoS

When you connect to an RDS DB instance through a public network, there may be risks of a distributed denial-of-service (DDoS) attack. If the RDS security system detects a DDoS attack, it will enable the anti-DDoS function. If the function cannot defend against the attack or the attack reaches the black hole threshold, black hole processing is triggered to ensure availability of the RDS service.

Security Protection

RDS is protected by multiple layers of firewalls to defend against various malicious attacks, such as DDoS attacks and SQL injections. For security reasons, you are advised to access RDS through a private network.

1.3.3 High Reliability

Dual-Host Hot Standby

RDS uses the hot standby architecture, in which failover upon fault occurrence takes only some seconds.

Data Backup

RDS automatically backs up data every day and transfers backup files to Object Storage Service (OBS). The backup files can be stored for 732 days and can be restored with just a few clicks. You can set a custom backup policy and create manual backups at any time.

Data Restoration

You can restore data from backups or to any point in time during the backup retention period. In most scenarios, you can use backup files to restore data to a new DB instance at any time point within 732 days. After the data is verified, data can be migrated back to the primary DB instance.

1.3.4 Comparison Between RDS and Self-Built Databases

Performance

Item	Cloud Database RDS	Self-Built Database Service
Service availability	For details, see section Elastic Cloud Service User Guide.	Requires device procurement, primary/standby relationship setup, and RAID setup.

Item	Cloud Database RDS	Self-Built Database Service
Data reliability	For more information, see the <i>Elastic Volume Service User Guide</i> .	Requires device procurement, primary/standby relationship setup, and RAID setup.
System security	Defends against Anti-DDoS attacks and promptly repairs database security vulnerabilities.	Requires procurement of expensive devices and software, as well as manual detection and repair of security vulnerabilities.
Database backup	Supports automated backups, manual backups, and custom backup retention periods.	Requires device procurement, setup, and maintenance.
Hardware and software investment	Supports on-demand pricing and scaling without requiring hardware and software investment.	Requires large investment in database servers.
System hosting	Not required.	Requires two servers for primary/standby DB instances.
Maintenance cost	Not required.	Requires large labor investment and professional database administrator (DBA) for maintenance.
Deployment and scaling	Supports elastic scaling, fast upgrade, and on-demand enabling.	Requires procurement, deployment, and coordination of hardware that matches original devices.
Resource utilization	Bills users based on the resources actually used, resulting in 100% resource utilization.	Considers peak traffic, resulting in low resource utilization.

1.4 Product Series

1.4.1 DB Instance Introduction

Currently, RDS DB instances are classified into the following types:

- Single
- Primary/Standby

Different series support different DB engines and instance specifications.

Table 1-1 DB instance types

DB Instance Type	Description	Scenarios
Single	Uses a single-node architecture. More cost-effective than the mainstream primary/standby DB instances.	<ul style="list-style-type: none"> • Personal learning • Microsites • Development and testing environment of small- and medium-sized enterprises
Primary/Standby	Uses an HA architecture with one master node and one slave node.	<ul style="list-style-type: none"> • Production databases of large- and medium-sized enterprises • Applications for the Internet, Internet of Things (IoT), retail e-commerce sales, logistics, gaming, and other industries

1.4.2 Function Comparison

Single DB instances use the single-node architecture. In contrast to the mainstream primary/standby DB instances, a single DB instance contains only one node and has no slave node for fault recovery.

Advantage Comparison

- Single DB instances: supports the creation of read replicas (except for Microsoft SQL Server) and supports the queries of error logs and slow query logs. Different from primary/standby DB instances that have two database nodes, a single DB instance has only one node. If a node fails, the restoration will take a long time. Therefore, single DB instances are not recommended for sensitive services that have high requirements on database availability.
- Primary/Standby DB instances: uses the slave database node only for failover and restoration. The slave database node does not provide services. The performance of single DB instances is similar to or even higher than the primary/standby DB instances.

Table 1-2 Function comparisons

DB Engine	Single	Primary/Standby	
MySQL	8.0	8.0	
	5.7	5.7	
	5.6	5.6	

DB Engine	Single	Primary/Standby	
PostgreSQL	11 10 9.6 9.5	11 10 9.6 9.5	
Number of nodes	1	2	
Specifications	vCPUs: a maximum of 76 Memory: a maximum of 304 GB Storage space: a maximum of 4 TB	vCPUs: a maximum of 76 Memory: a maximum of 304 GB Storage space: a maximum of 4 TB	
Monitoring and alarms	Supported	Supported	
Security group	Supported	Supported	
Backups and restorations	Supported	Supported	
Parameter settings	Supported	Supported	
SSL	Supported	Supported	
Log management	Supported	Supported	

DB Engine	Single	Primary/Standby	
Read replicas (need to be created)	Supported	Supported	
High-frequency monitoring	Supported	Supported	
Primary/standby switch over or failover	Not supported	Supported	
Standby DB instance migration	Not supported	Supported	

1.5 DB Instance Description

1.5.1 DB Instance Types

DB instances are the smallest management units used by RDS. A DB instance is an isolated database environment on the cloud. A DB instance can contain multiple user-created databases, and you can access it using the same tools and applications that you use with a stand-alone DB instance. You can create and modify DB instances using the management console or APIs. RDS does not have limits on the number of running DB instances. Each DB instance has a DB instance identifier.

DB instances are classified into the following types.

Table 1-3 DB instance types

DB Instance Type	Description
Single	Uses a single-node architecture. More cost-effective than primary/standby DB instances.
Primary/Standby	Uses an HA architecture with one master node and one slave node.
Read replica	Uses a single-node architecture (without a standby node).

You can use RDS to create and manage DB instances running various DB engines.

For details about differences and function comparison between different instance types, see [DB Instance Introduction](#) and [Function Comparison](#).

1.5.2 DB Instance Classes

General-enhanced and general-enhanced II instance classes provide robust and stable performance. They use latest-generation network acceleration engines and Data Plane Development Kit (DPDK) to provide higher network performance, meeting requirements in different scenarios.

- General-enhanced DB instances use Intel® Xeon® Scalable processors and feature high and stable computing performance. Working in high-performance networks, general-enhanced DB instances provide higher performance and stability, meeting enterprise-class application requirements.
- General-enhanced II DB instances use second-generation Intel® Xeon® Scalable processors with technologies optimized and 25GE high-speed intelligent NICs to offer powerful and stable computing performance, including ultra-high network bandwidth and PPS.

Table 1-4 DB instance classes

Instance Class	vCPUs	Memory (GB)	Supported DB Engine
General-enhanced	1	2	MySQL
	2	4	MySQL
	2	8	MySQL
	4	8	MySQL
	4	16	MySQL
	4	32	MySQL
	8	16	MySQL
	8	32	MySQL

Instance Class	vCPUs	Memory (GB)	Supported DB Engine
	16	32	<ul style="list-style-type: none"> MySQL PostgreSQL
	16	64	<ul style="list-style-type: none"> MySQL PostgreSQL
	32	128	<ul style="list-style-type: none"> MySQL PostgreSQL
General-enhanced II	2	4	MySQL
	2	8	MySQL
	4	8	MySQL
	4	16	MySQL
	8	16	MySQL
	8	32	MySQL
	8	64	MySQL
	16	32	MySQL
	16	64	<ul style="list-style-type: none"> MySQL PostgreSQL
	32	64	<ul style="list-style-type: none"> MySQL PostgreSQL
	32	128	PostgreSQL
	64	128	<ul style="list-style-type: none"> MySQL PostgreSQL
	64	256	<ul style="list-style-type: none"> MySQL PostgreSQL

The DB instance specifications vary according to site requirements.

1.5.3 DB Engines and Versions

Table 1-5 lists the DB engines and versions supported by RDS.

For new applications, you are advised to use the latest major version of the DB engine, for example, MySQL 5.7. When you create a DB instance, you can select a major DB engine version only (such as MySQL 5.7). The system will automatically select an appropriate minor version (such as 5.7.23). After the DB instance is created, you can view the minor version in the **DB Engine Version** column on the

Instance Management page. The DB engine and version vary according to site requirements.

Table 1-5 DB engines and versions

DB Engine	Version
MySQL	<ul style="list-style-type: none"> • 8.0 • 5.7 • 5.6
PostgreSQL	<ul style="list-style-type: none"> • 11 • 10 • 9.6 • 9.5

1.5.4 DB Instance Statuses

DB Instance Statuses

The status of a DB instance indicates the health of the DB instance. You can use the management console or API to view the status of a DB instance.

Table 1-6 DB instance statuses

Status	Description
Available	DB instance is available.
Abnormal	DB instance is abnormal.
Creating	DB instance or backup is being created.
Creation failed	DB instance has failed to be created.
Switchover in progress	Standby DB instance is being switched over to the primary DB instance.
Changing type to primary/standby	Single DB instance is being changed to primary/standby DB instances.
Rebooting	DB instance is being rebooted.
Changing port	DB instance port is being changed.
Changing instance class	CPU or memory of a DB instance is being modified.
Scaling up	Storage space of a DB instance is being scaled up.
Restoring	DB instance is being restored from a backup.

Status	Description
Restore failed	DB instance fails to be restored.
Storage full	Storage space of the DB instance is full. Data cannot be written to databases.
Deleted	DB instance has been deleted and will not be displayed in the instance list.
Pending reboot	A modification to a database parameter is waiting for an instance reboot before it can take effect.

1.6 Typical Applications

1.6.1 Read/Write Splitting

To improve the system processing capability, you can simply create read replicas without the need of changing your existing applications.

1.7 Constraints

1.7.1 MySQL Constraints

Table 1-7 shows the constraints designed to ensure the stability and security of RDS for MySQL.

Table 1-7 Function constraints

Function Item	Constraints
Database access	<ul style="list-style-type: none"> • If public accessibility is not enabled for the RDS DB instance, the RDS DB instance must be in the same VPC as the ECS. • RDS read replicas must be created in the same subnet as the primary DB instance. • The security group must allow access from an ECS. By default, RDS cannot be accessed through an ECS in a different security group. You need to add an inbound rule to the RDS security group. • The default RDS port is 3306. You can change it if you want to access RDS through another port.
Deployment	ECSs in which DB instances are deployed are not visible to users. You can access the DB instances only through an IP address and a port number.

Function Item	Constraints
Database root permissions	Only the root user permissions are provided on the instance creation page.
Database parameter modification	Most parameters can be modified on the RDS console.
Data import	<ul style="list-style-type: none"> Through the command-line interface (CLI) or graphical user interface (GUI) Through MySQL CLI tools For details, see Migrating Data to RDS for MySQL Using mysqldump .
MySQL storage engine	For details, see What Storage Engines Does the RDS for MySQL Support?
Database replication setup	RDS for MySQL provides a dual-node cluster with the primary/standby replication architecture. You do not need to set up replication. The standby DB instance is not visible to users and therefore you cannot access it directly.
DB instance reboot	RDS DB instances cannot be rebooted through commands. They must be rebooted on the RDS console.
RDS backup files	RDS backup files are stored in OBS buckets and are not visible to users.

1.7.2 PostgreSQL Constraints

Table 1-8 shows the constraints designed to ensure the stability and security of RDS for PostgreSQL.

Table 1-8 Function constraints

Function Item	Constraints
Database access	<ul style="list-style-type: none"> If public accessibility is not enabled for the RDS DB instance, the RDS DB instance must be in the same VPC as the ECS. RDS read replicas must be created in the same subnet as the primary DB instance. The security group must allow access from an ECS. By default, RDS cannot be accessed through an ECS in a different security group. You need to add an inbound rule to the RDS security group. The default RDS port is 5432. You can change it if you want to access RDS through another port.

Function Item	Constraints
Deployment	ECSs in which DB instances are deployed are not visible to users. You can access the DB instances only through an IP address and a port number.
Database root permissions	Only the root user permissions are provided on the instance creation page.
Database parameter modification	Most parameters can be modified on the RDS console.
Data import	<ul style="list-style-type: none"> Through the psql CLI tools For details, see section Migrating Data to RDS for PostgreSQL Using psql .
Database replication setup	RDS for PostgreSQL provides a dual-node cluster with the primary/standby replication architecture. You do not need to set up replication. The standby DB instance is not visible to users and therefore you cannot access it directly.
DB instance reboot	DB instances cannot be rebooted through commands. They must be rebooted on the RDS console.
RDS backup files	RDS backup files are stored in OBS buckets and are not visible to users.

1.8 Related Services

Table 1-9 Related services

Service Name	Description
Elastic Cloud Service (ECS)	Enables you to access RDS DB instances through an ECS to reduce application response time and public network traffic fees.
Virtual Private Cloud (VPC)	Isolates your networks and controls access to your RDS DB instances.
Object Storage Service (OBS)	Stores automated and manual backups of your RDS DB instances.
Cloud Eye	Acts as an open monitoring platform that monitors RDS resources in real time. It reports alarms and issues warnings promptly to ensure that services are running properly.

2 Getting Started with RDS for MySQL

2.1 Connecting to a DB Instance

An RDS DB instance can be connected through a private network or a public network.

Table 2-1 RDS connection methods

Connect Through	IP Address	Scenarios	Description
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.	<ul style="list-style-type: none"> • Secure and excellent performance • Recommended

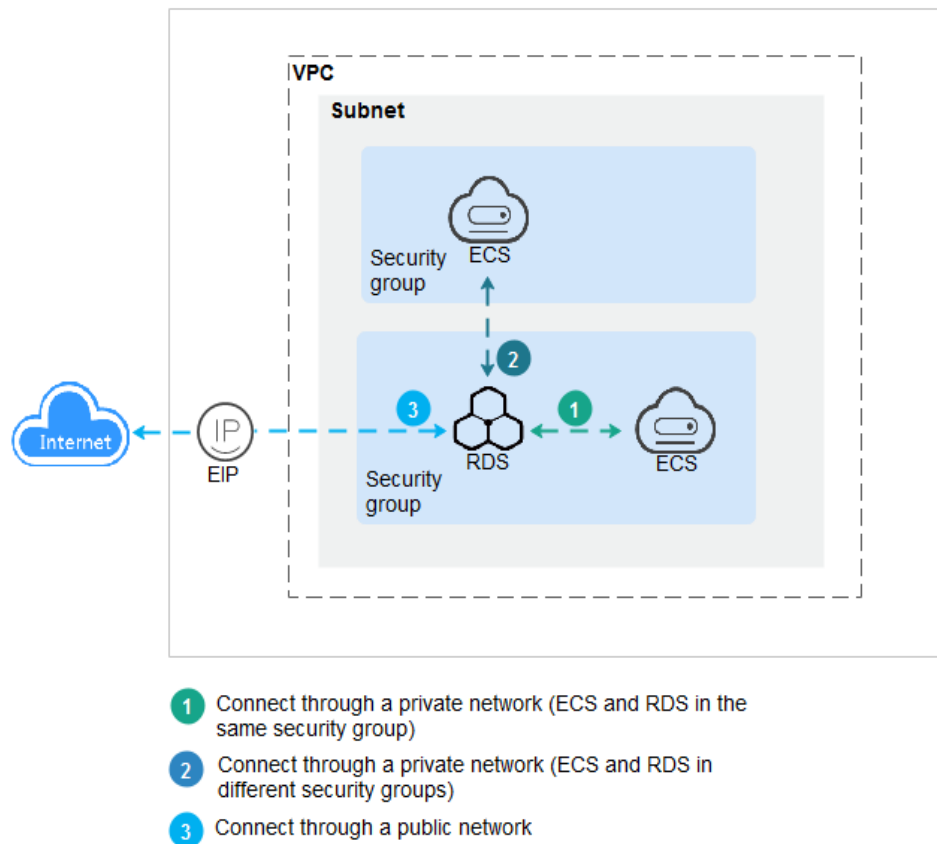
Connect Through	IP Address	Scenarios	Description
Public network	EIP	If you cannot access an RDS DB instance through a floating IP address, bind an EIP to the DB instance and connect the DB instance to the ECS through the EIP.	<ul style="list-style-type: none"> • A relatively lower level of security compared 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.

 **NOTE**

- VPC: indicates the Virtual Private Cloud.
- ECS: indicates the 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 2-1 illustrates the connection over a private network or a public network.

Figure 2-1 DB instance connection



2.2 Connecting to a MySQL DB Instance Through a Private Network

2.2.1 Overview

Scenarios

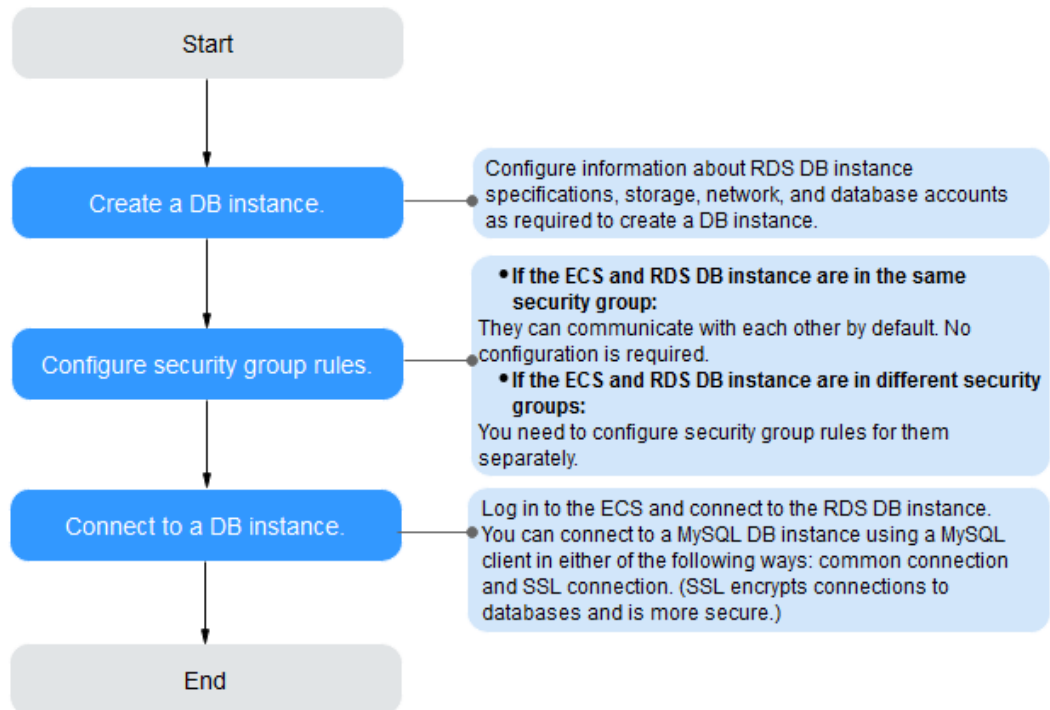
You can create a DB instance on the management console and connect to the DB instance through an ECS.

If you are using RDS for the first time, see the constraints described in section [MySQL Constraints](#).

Process

[Figure 2-2](#) illustrates the process of connecting to a MySQL DB instance through a private network.

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



- **Step 1: Create a DB instance.** Confirm the specifications, storage, network, and database account configurations of the MySQL DB instances based on service requirements.
- **Step 2: Configure security group rules.**
 - 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 **Step 3: Connect to a DB Instance Through a Private Network**.
 - 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 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.
- **Step 3: Connect to a DB instance through a private network.** You can connect to a DB instance through a common connection or an SSL connection. The SSL connection encrypts data and is more secure.

2.2.2 Step 1: Create a DB Instance

Scenarios

This section describes how to create a DB instance on the management console.

The DB instance class and storage space you need depend on your processing power and memory requirements.

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 **Service List**. Under **Database**, click **Relational Database Service**. The RDS console is displayed.
- Step 4** On the **Instance Management** page, click **Create DB Instance**.
- Step 5** On the displayed page, configure information about your DB instance. Then, click **Create Now**.

Table 2-2 Basic information

Parameter	Description
Region	The region your RDS resources will be located. You can change it on the creation page, or go back to the Instance Management page and change it in the upper left corner. 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 Instance Name	Must start with a letter and consist of 4 to 64 characters. Only letters, digits, hyphens (-), and underscores (_) are allowed.
DB Engine	Set to MySQL .
DB Engine Version	For details, see DB Engines and Versions . Different DB engine versions are supported in different regions. If you use a MySQL database, select a proper DB engine version based on service requirements. You are advised to select the latest available version because it is more stable, reliable, and secure.

Parameter	Description
DB Instance Type	<ul style="list-style-type: none"> ● Primary/Standby: uses the 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. An AZ is a physical region where resources use independent power supply and networks. AZs are physically isolated but interconnected through an internal network. RDS allows you to deploy primary/standby DB instances in an AZ or across AZs. You can determine whether the standby AZ is the same as the primary AZ. <ul style="list-style-type: none"> - If they are the same (default setting), the primary and standby DB instances are deployed in the same AZ. - If they are different, the primary and standby DB instances are deployed in different AZs to ensure failover support and high availability. ● Single: uses the single-node architecture, which is more cost-effective than mainstream primary/standby DB instances. It is suitable for developing and testing of microsites, and small- and medium-sized enterprises, or for learning about RDS.
Storage Type	<p>Determines the DB instance read/write speed. The higher the maximum throughput is, the higher the DB instance read/write speed can be.</p> <ul style="list-style-type: none"> ● Common I/O: uses the SATA disk type that supports a maximum throughput of 90 MB/s. ● Ultra-high I/O: supports a maximum throughput of 350 MB/s.
Time Zone	<p>Select a time zone when you are creating a DB instance, and you can change it after the DB instance is created.</p>

Table 2-3 Instance specifications

Parameter	Description
Instance Class	<p>Refers to the CPU and memory of a DB instance. Different instance classes have different numbers of database connections and different maximum IOPS.</p> <p>For details about instance classes, see section DB Instance Classes.</p> <p>After a DB instance is created, you can change its instance class. For details, see section Changing a DB Instance Class.</p>
Storage Type	<p>Determines the DB instance read/write speed. The higher the maximum throughput is, the higher the DB instance read/write speed can be.</p> <ul style="list-style-type: none"> • Common I/O: uses the SATA disk type, with a maximum throughput of 90 MB/s. • Ultra-high I/O: supports a maximum throughput of 350 MB/s.
Storage Space (GB)	<p>Contains the file system overhead required for inode, reserved block, 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.</p> <p>After a DB instance is created, you can scale up its storage space. For details, see section Scaling Up Storage Space.</p>

Table 2-4 Network

Parameter	Description
VPC	<p>A dedicated virtual network in which your RDS DB instances are located. A VPC can isolate networks for different services. You can select an existing VPC or create a VPC. For details on how to create a VPC, see the "Creating a VPC" section in the <i>Virtual Private Cloud User Guide</i>.</p> <p>If no VPC is available, RDS allocates a VPC for you by default.</p> <p>NOTICE After the DB instance is created, the VPC cannot be changed.</p>

Parameter	Description
Subnet	<p>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.</p> <p>A floating IP address is automatically assigned when you create a DB instance. You can also enter an unused floating IP address in the subnet CIDR block. After the DB instance is created, you can change the floating IP address.</p>
Security Group	<p>Enhances security by controlling access to RDS from other services. You need to add rules to a security group so that you can connect to your DB instance.</p> <p>If no security group is available or has been created, RDS allocates a security group to you by default.</p>

Table 2-5 Database configuration

Parameter	Description
Administrator	The default login name for the database is root .
Administrator Password	<p>Must consist of 8 to 32 characters and contain at least three of the following: uppercase letters, lowercase letters, digits, and special characters (~!@#%^*_+?,). Enter a strong password. Periodically change it to improve security and prevent security risks such as brute force cracking.</p> <p>Keep this password secure. The system cannot retrieve it.</p> <p>After a DB instance is created, you can reset this password. For details, see section Resetting the Administrator Password.</p>
Confirm Password	Must be the same as Administrator Password .

Parameter	Description
Parameter Template	<p>Contains engine configuration values that can be applied to one or more DB instances. If you intend to create primary/standby DB instances, they use the same parameter template. You can modify the instance parameters as required after the DB instance is created.</p> <p>NOTICE</p> <p>If you use a custom parameter template when creating a DB instance, the following specification-related parameters in the custom template are not delivered. Instead, the default values are used.</p> <ul style="list-style-type: none"> • back_log • innodb_io_capacity_max • max_connections • innodb_io_capacity • innodb_buffer_pool_size • innodb_buffer_pool_instances <p>You can modify the instance parameters as required after the DB instance is created. For details, see section Modifying Parameters.</p>

Table 2-6 Tags

Parameter	Description
Tag	<p>Tags an RDS DB instance. This configuration is optional. Adding tags to RDS DB instances helps you better identify and manage the DB instances. A maximum of 10 tags can be added for each DB instance.</p>

Table 2-7 Batch creation

Parameter	Description
Quantity	<p>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.</p> <p>If you create multiple DB instances at a time, they will be named with four digits appended to the DB instance name. For example, if you enter instance, the first instance will be named as instance-0001, the second as instance-0002, and so on.</p>

 **NOTE**


The performance of your DB instance depends on its configurations. Hardware configuration items include the instance specifications, storage type, and storage space.

Step 6 Confirm the specifications.

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

Step 7 To view and manage the DB instance, go to the **Instance Management** page.

- During the creation process, the DB instance status is **Creating**. You can view the detailed progress and result of the task on the **Task Center** page. To

refresh the DB instance list, click  in the upper right corner of the list. When the creation process is complete, the instance status will change to **Available**.

- The automated backup policy is enabled by default. After the DB instance is created, you can modify the automated backup policy. An automated full backup is immediately triggered after a DB instance is created.
- The default database port is **3306**. After a DB instance is created, you can change its port.

For details, see section [Changing the Database Port](#).

----End

2.2.3 Step 2: Configure 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 in a VPC.

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

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 [Step 3: Connect to a DB Instance Through a Private Network](#).
- 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 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.

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 500 security group rules.
- Too many security group rules will increase the first packet latency. You are advised to create a maximum of 50 rules for each security group.
- To access 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

If you use **0.0.0.0/0**, you enable all IP addresses to access RDS DB instances in the security group.

Procedure

- Step 1** Log in to the management console.
- Step 2** Under **Network**, click **Virtual Private Cloud**.
- Step 3** In the navigation pane on the left, choose **Access Control > Security Groups**.
- Step 4** On the **Security Groups** page, locate the target security group and click **Manage Rule** in the **Operation** column.
- Step 5** On the displayed page, click **Add Rule**.
- Step 6** In the displayed dialog box, set required parameters to add an inbound rule.
- Step 7** Click **OK**.

----End

2.2.4 Step 3: Connect to a DB Instance Through a Private Network

You can connect to a DB instance through a common connection or an SSL connection. The SSL connection encrypts data and is more secure.

Prerequisites

1. Log in to the ECS.
 - To connect to a DB instance through an ECS, you must ensure that:
 - The ECS and DB instance must be in the same VPC.
 - The ECS must be allowed by the security group to access RDS DB instances.
 - If the security group to which the target DB instance belongs is the default security group, you do not need to configure security group rules.

- If the security group to which the target DB instance belongs is not the default security group, check whether the security group rules allow the ECS to connect to the DB instance. For details, see section [Step 2: Configure Security Group Rules](#).

If the security group rules allow the access from the ECS, the ECS can connect to the DB instance.

If the security group rules do not allow the access from the ECS, you need to add a security group rule. The ECS must be allowed by the security group to access RDS DB instances.

2. Use a database client to connect to the target DB instance.

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

- In the Linux OS, install the MySQL client on the device that can access RDS. It is recommended that you download a MySQL client running a version later than that of the DB instance.

For details about how to obtain and install the MySQL client, see section [How Can I Install the MySQL Client?](#)

- In the Windows OS, you can use any common database client to connect to the target DB instance in a similar way.

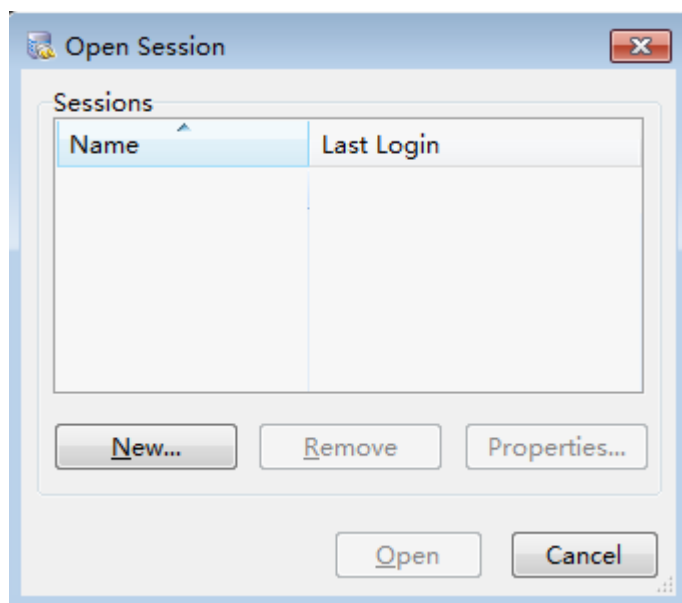
The database client MySQL-Front is used as an example in [Using MySQL-Front to Connect to a DB Instance](#).

Using MySQL-Front to Connect to a DB Instance

Step 1 Start MySQL-Front.

Step 2 In the displayed dialog box, click **New**.

Figure 2-3 Connection management



Step 3 Enter the information of the DB instance to be connected and click **Ok**, as shown in [Figure 2-4](#).

Figure 2-4 Adding an account

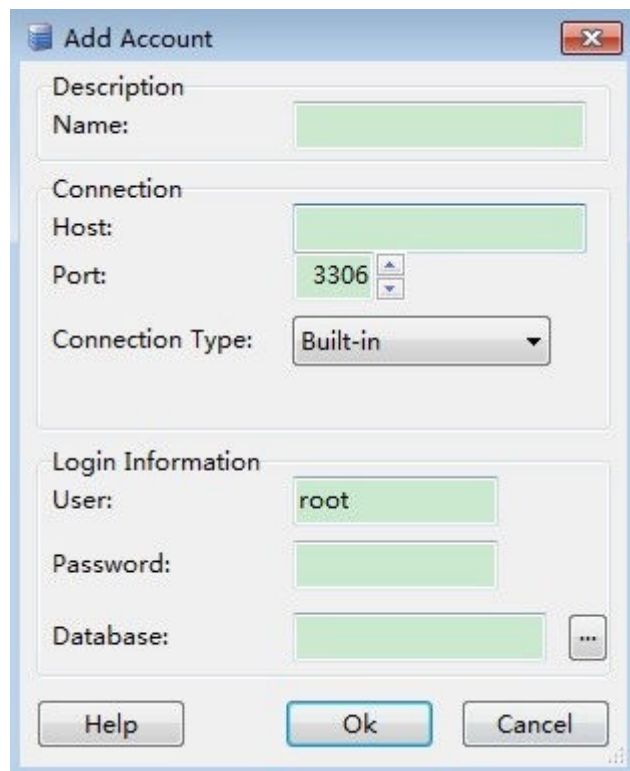
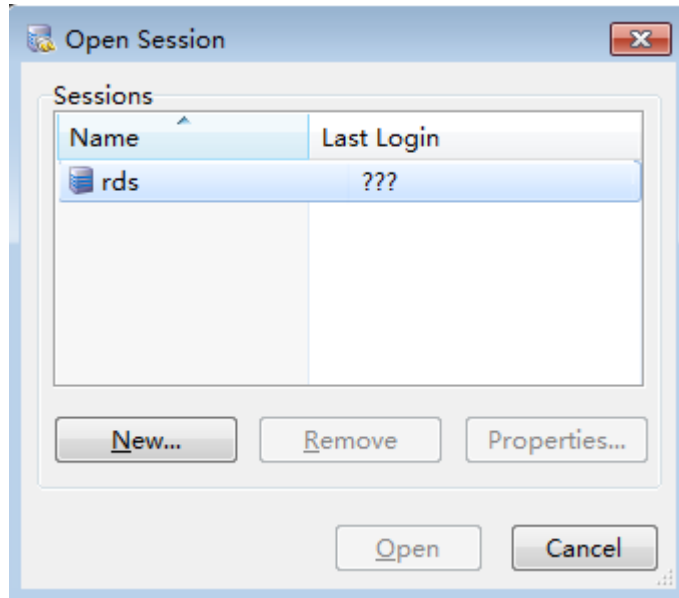


Table 2-8 Parameter description

Parameter	Description
Name	Indicates the name of the database connection task. If you do not set this parameter, it will be the same as Host by default.
Host	Indicates the floating IP address of the DB instance to be connected. To view the floating IP address and port of the DB instance, perform the following steps: <ol style="list-style-type: none"> 1. Log in to the RDS console. 2. Select the region in which the DB instance is located. 3. Click the target DB instance to enter the Basic Information page. 4. In the Connection Information area, view the floating IP address.
Port	Indicates the private network port of the DB instance.
User	Indicates the name of the user who will access the DB instance. The default user is root .
Password	Indicates the password of the RDS database account.

- Step 4** In the displayed window, select the connection that you have created in [Step 3](#) and click **Open**. If the connection information is correct, the DB instance is successfully connected.

Figure 2-5 Opening a session





NOTE

If the connection fails, see [What Should I Do If an ECS Cannot Connect to an RDS DB Instance?](#)

----End

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 **Service List**. Under **Database**, click **Relational Database Service**. The RDS console is displayed.
- Step 4** In the **DB Information** area on the **Basic Information** page, click  in the **SSL** field to download the root certificate or certificate bundle.
- Step 5** Import the root certificate to the Linux OS on the ECS. For details, see [How Can I Import the Root Certificate to the Windows or Linux OS?](#)

 **NOTE**

- Since April 2017, RDS has offered a new root certificate that has a 20-year validation period. The new certificate takes effect after DB instances are rebooted. Replace the old certificate before it expires to improve system security.

For details, see section [How Can I Identify the Validity Period of an SSL Root Certificate?](#)

- You can also download the certificate bundle, which contains both the new certificate provided since April 2017 and the old certificate.

Step 6 Connect to an RDS DB instance. The Linux OS is used as an example.

```
mysql -h <hostName> -P <port> -u <userName> -p --ssl-ca=<caName>
```

Table 2-9 Parameter description

Parameter	Description
<hostName>	Indicates the floating IP address. To obtain this parameter, go to the Basic Information page of the DB instance and view the floating IP address in the Connection Information area.
<port>	Indicates the database port. By default, the value is 3306 . To obtain this parameter, go to the Basic Information page of the DB instance and view the database port in the Connection Information area.
<userName>	Indicates the username of the RDS database account. The default administrator is root .
<caName>	Indicates the name of the SSL certificate file, which should be stored in the same directory where the command is executed.

For example, to connect to a DB instance through an SSL connection as user **root**, run the following command:

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

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

Enter password:

 **NOTE**

If the connection fails, see [What Should I Do If an ECS Cannot Connect to an RDS DB Instance?](#)

----End

2.3 Connecting to a MySQL DB Instance Through a Public Network

2.3.1 Overview

Scenarios

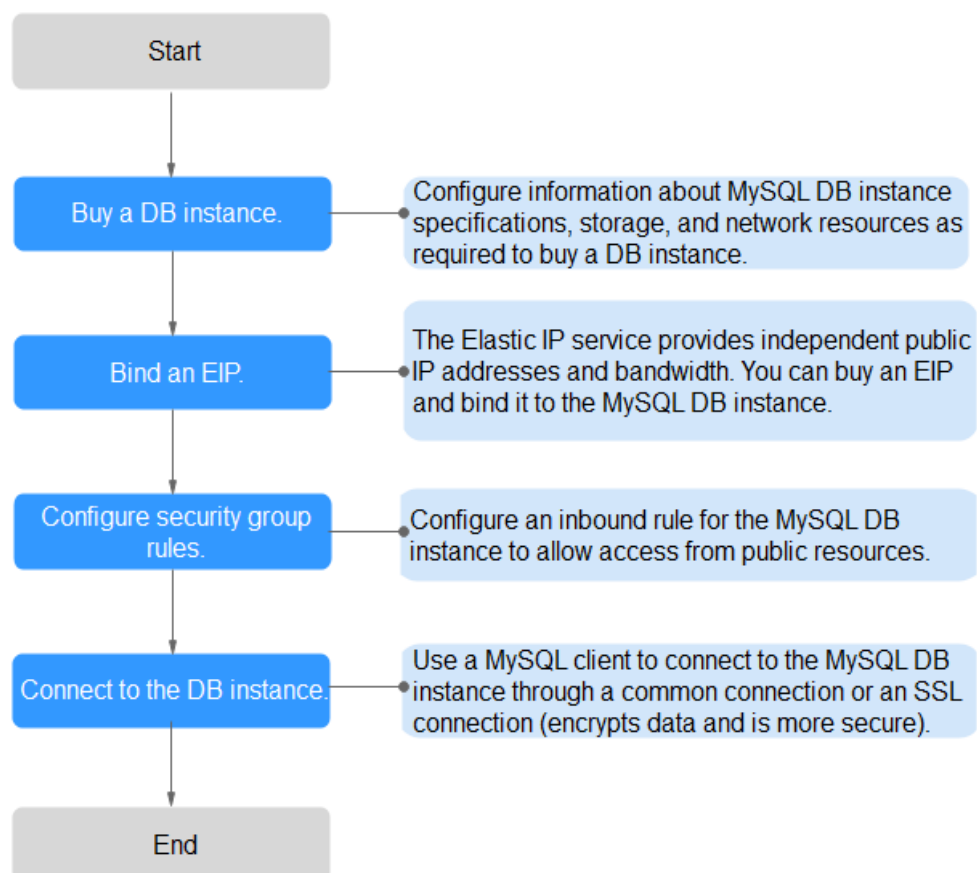
This section describes how to create a MySQL DB instance on the management console and bind an EIP to the DB instance to make the instance publicly accessible.

If you are using RDS for the first time, see the constraints described in section [MySQL Constraints](#).

Process

Figure 2-6 illustrates the process of connecting to a MySQL DB instance through a public network.

Figure 2-6 Connecting to a DB instance through a public network



- **Step 1: Create a DB instance.** Confirm the specifications, storage, network, and database account configurations of the MySQL DB instances based on service requirements.
- **Step 2: Bind an EIP.** The EIP provides independent public IP addresses and bandwidth for Internet access. You can apply for an EIP on the VPC console and bind the EIP to the RDS DB instance.
- **Step 2: Configure security group rules.** To access 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.
- **Step 4: Connect to a DB instance through a public network.** You can connect to a DB instance through a common connection or an SSL connection. The SSL connection encrypts data and is more secure.

2.3.2 Step 1: Create a DB Instance

Scenarios

This section describes how to create a DB instance on the management console.

The DB instance class and storage space you need depend on your processing power and memory requirements.

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 **Service List**. Under **Database**, click **Relational Database Service**. The RDS console is displayed.
- Step 4** On the **Instance Management** page, click **Create DB Instance**.
- Step 5** On the displayed page, configure information about your DB instance. Then, click **Create Now**.

Table 2-10 Basic information

Parameter	Description
Region	The region your RDS resources will be located. You can change it on the creation page, or go back to the Instance Management page and change it in the upper left corner. 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 Instance Name	Must start with a letter and consist of 4 to 64 characters. Only letters, digits, hyphens (-), and underscores (_) are allowed.
DB Engine	Set to MySQL .

Parameter	Description
DB Engine Version	<p>For details, see DB Engines and Versions.</p> <p>Different DB engine versions are supported in different regions.</p> <p>If you use a MySQL database, select a proper DB engine version based on service requirements. You are advised to select the latest available version because it is more stable, reliable, and secure.</p>
DB Instance Type	<ul style="list-style-type: none"> ● Primary/Standby: uses the 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. An AZ is a physical region where resources use independent power supply and networks. AZs are physically isolated but interconnected through an internal network. RDS allows you to deploy primary/standby DB instances in an AZ or across AZs. You can determine whether the standby AZ is the same as the primary AZ. <ul style="list-style-type: none"> - If they are the same (default setting), the primary and standby DB instances are deployed in the same AZ. - If they are different, the primary and standby DB instances are deployed in different AZs to ensure failover support and high availability. ● Single: uses the single-node architecture, which is more cost-effective than mainstream primary/standby DB instances. It is suitable for developing and testing of microsites, and small- and medium-sized enterprises, or for learning about RDS.
Storage Type	<p>Determines the DB instance read/write speed. The higher the maximum throughput is, the higher the DB instance read/write speed can be.</p> <ul style="list-style-type: none"> ● Common I/O: uses the SATA disk type that supports a maximum throughput of 90 MB/s. ● Ultra-high I/O: supports a maximum throughput of 350 MB/s.
Time Zone	<p>Select a time zone when you are creating a DB instance, and you can change it after the DB instance is created.</p>

Table 2-11 Instance specifications

Parameter	Description
Instance Class	<p>Refers to the CPU and memory of a DB instance. Different instance classes have different numbers of database connections and different maximum IOPS.</p> <p>For details about instance classes, see section DB Instance Classes.</p> <p>After a DB instance is created, you can change its instance class. For details, see section Changing a DB Instance Class.</p>
Storage Type	<p>Determines the DB instance read/write speed. The higher the maximum throughput is, the higher the DB instance read/write speed can be.</p> <ul style="list-style-type: none"> • Common I/O: uses the SATA disk type, with a maximum throughput of 90 MB/s. • Ultra-high I/O: supports a maximum throughput of 350 MB/s.
Storage Space (GB)	<p>Contains the file system overhead required for inode, reserved block, 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.</p> <p>After a DB instance is created, you can scale up its storage space. For details, see section Scaling Up Storage Space.</p>

Table 2-12 Network

Parameter	Description
VPC	<p>A dedicated virtual network in which your RDS DB instances are located. A VPC can isolate networks for different services. You can select an existing VPC or create a VPC. For details on how to create a VPC, see the "Creating a VPC" section in the <i>Virtual Private Cloud User Guide</i>.</p> <p>If no VPC is available, RDS allocates a VPC for you by default.</p> <p>NOTICE After the DB instance is created, the VPC cannot be changed.</p>

Parameter	Description
Subnet	<p>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.</p> <p>A floating IP address is automatically assigned when you create a DB instance. You can also enter an unused floating IP address in the subnet CIDR block. After the DB instance is created, you can change the floating IP address.</p>
Security Group	<p>Enhances security by controlling access to RDS from other services. You need to add rules to a security group so that you can connect to your DB instance.</p> <p>If no security group is available or has been created, RDS allocates a security group to you by default.</p>

Table 2-13 Database configuration

Parameter	Description
Administrator	The default login name for the database is root .
Administrator Password	<p>Must consist of 8 to 32 characters and contain at least three of the following: uppercase letters, lowercase letters, digits, and special characters (~!@#%^*_+?,.). Enter a strong password. Periodically change it to improve security and prevent security risks such as brute force cracking.</p> <p>Keep this password secure. The system cannot retrieve it.</p> <p>After a DB instance is created, you can reset this password. For details, see section Resetting the Administrator Password.</p>
Confirm Password	Must be the same as Administrator Password .

Parameter	Description
Parameter Template	<p>Contains engine configuration values that can be applied to one or more DB instances. If you intend to create primary/standby DB instances, they use the same parameter template. You can modify the instance parameters as required after the DB instance is created.</p> <p>NOTICE</p> <p>If you use a custom parameter template when creating a DB instance, the following specification-related parameters in the custom template are not delivered. Instead, the default values are used.</p> <ul style="list-style-type: none"> • back_log • innodb_io_capacity_max • max_connections • innodb_io_capacity • innodb_buffer_pool_size • innodb_buffer_pool_instances <p>You can modify the instance parameters as required after the DB instance is created. For details, see section Modifying Parameters.</p>

Table 2-14 Tags

Parameter	Description
Tag	<p>Tags an RDS DB instance. This configuration is optional. Adding tags to RDS DB instances helps you better identify and manage the DB instances. A maximum of 10 tags can be added for each DB instance.</p>

Table 2-15 Batch creation

Parameter	Description
Quantity	<p>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.</p> <p>If you create multiple DB instances at a time, they will be named with four digits appended to the DB instance name. For example, if you enter instance, the first instance will be named as instance-0001, the second as instance-0002, and so on.</p>

 **NOTE**


The performance of your DB instance depends on its configurations. Hardware configuration items include the instance specifications, storage type, and storage space.

Step 6 Confirm the specifications.

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

Step 7 To view and manage the DB instance, go to the **Instance Management** page.

- During the creation process, the DB instance status is **Creating**. You can view the detailed progress and result of the task on the **Task Center** page. To

refresh the DB instance list, click  in the upper right corner of the list. When the creation process is complete, the instance status will change to **Available**.

- The automated backup policy is enabled by default. After the DB instance is created, you can modify the automated backup policy. An automated full backup is immediately triggered after a DB instance is created.
- The default database port is **3306**. After a DB instance is created, you can change its port.

For details, see section [Changing the Database Port](#).

----End

2.3.3 Step 2: Bind an EIP

Scenarios

By default, a DB instance is not publicly accessible (not bound with an EIP) after being created. You can bind an EIP to the DB instance for public access 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 [Step 3: Configure Security Group Rules](#).
- Public accessibility reduces the security of DB instances. Therefore, exercise caution when enabling this function. To achieve a higher transmission rate and security level, you are advised to migrate your applications to the ECS that is in the same region as RDS.

Binding an EIP

Step 1 On the **Instance Management** page, click the target DB instance.

Step 2 In the navigation pane on the left, choose **EIPs**. On the displayed page, click **Bind EIP**.

Step 3 In the displayed dialog box, select an EIP and click **OK**.

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

Step 4 On the **EIPs** 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

2.3.4 Step 3: Configure 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 in 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 500 security group rules.
- Too many security group rules will increase the first packet latency. You are advised to create a maximum of 50 rules for each security group.
- To access 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

If you use **0.0.0.0/0**, you enable all IP addresses to access RDS DB instances in the security group.

Procedure

Step 1 Log in to the management console.

Step 2 Under **Network**, click **Virtual Private Cloud**.

Step 3 In the navigation pane on the left, choose **Access Control > Security Groups**.

Step 4 On the **Security Groups** page, locate the target security group and click **Manage Rule** in the **Operation** column.

Step 5 On the displayed page, click **Add Rule**.

Step 6 In the displayed dialog box, set required parameters to add an inbound rule.

Step 7 Click **OK**.

----End

2.3.5 Step 4: Connect to a DB Instance Through a Public Network

You can connect to a DB instance through a common connection or an SSL connection. The SSL connection encrypts data and is more secure.

Preparations

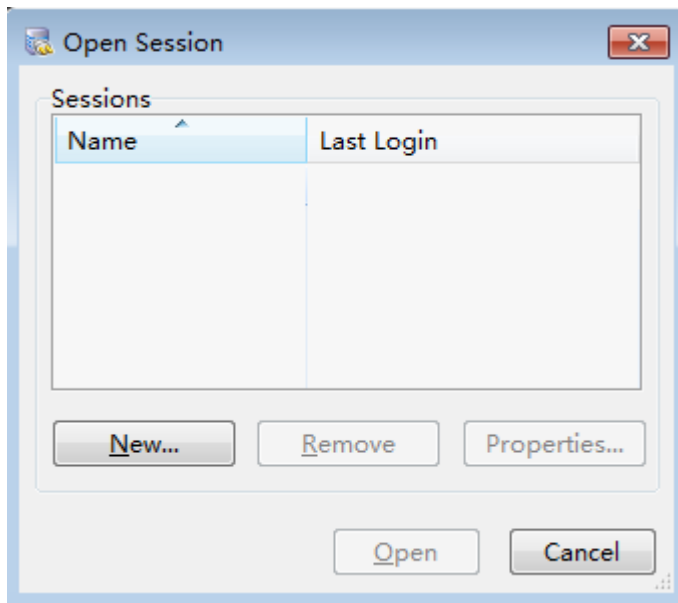
1. Bind an EIP to the target DB instance and configure security group rules.
 - a. Bind an EIP to the target DB instance.
For details about how to bind an EIP, see section [Step 2: Bind an EIP](#).
 - b. Obtain the IP address of the local device.
 - c. Configure security group rules.
Add the IP address and port obtained in [1.b](#) to the inbound rule of the security group.
For details about how to configure a security group rule, see section [Step 3: Configure Security Group Rules](#).
 - d. Run the **ping** command to check the connectivity between the local device and the DB instance.
2. Use a database client to connect to the target DB instance.
You can use a database client to connect to the target DB instance in the Linux or Windows operating system (OS).
 - In the Linux OS, you need to install a MySQL client on the ECS. It is recommended that you download a MySQL client running a version later than that of the DB instance.
For details about how to obtain and install the MySQL client, see section [How Can I Install the MySQL Client?](#)
 - In the Windows OS, you can use any common database client to connect to the target DB instance in a similar way.
The database client MySQL-Front is used as an example in [Using MySQL-Front to Connect to a DB Instance](#).

Using MySQL-Front to Connect to a DB Instance

Step 1 Start MySQL-Front.

Step 2 In the displayed dialog box, click **New**.

Figure 2-7 Connection management



Step 3 Enter the information of the DB instance to be connected and click **Ok**, as shown in [Figure 2-8](#).

Figure 2-8 Adding an account

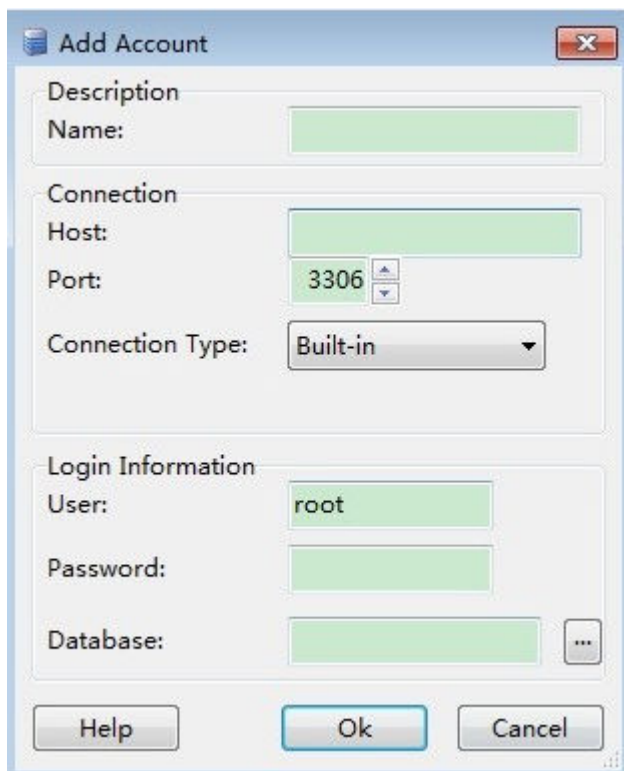


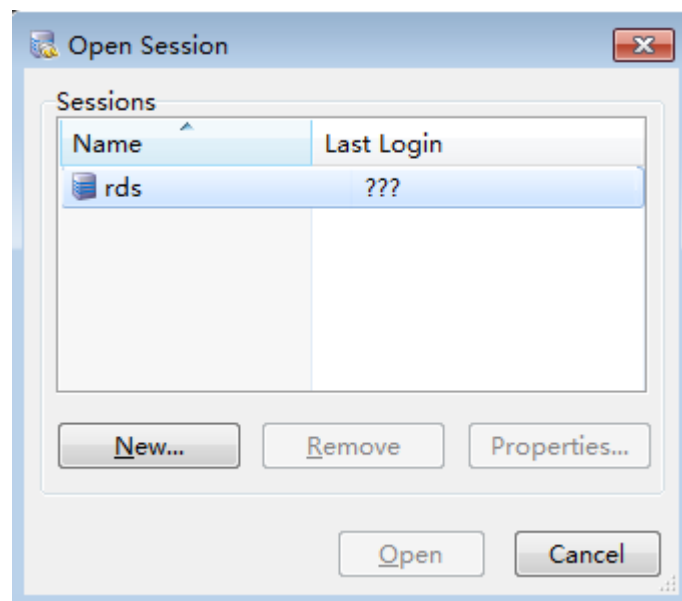
Table 2-16 Parameter description

Parameter	Description
Name	Indicates the name of the database connection task. If you do not set this parameter, it will be the same as Host by default.
Host	Indicates the EIP of the DB instance to be connected.
Port	Indicates the private network port of the DB instance.
User	Indicates the name of the user who will access the DB instance. The default user is root .
Password	Indicates the password of the RDS database account.

Step 4 In the displayed window, select the connection that you have created in [Figure 2-9](#) and click **Open**.

If the connection information is correct, the DB instance is successfully connected.

Figure 2-9 Opening a session





NOTE

If the connection fails, ensure that preparations have been correctly made in [Preparations](#) and try again.

----End

Using SSL to Connect to a DB Instance

- 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 **Service List**. Under **Database**, click **Relational Database Service**. The RDS console is displayed.
- Step 4** In the **DB Information** area on the **Basic Information** page, click  in the **SSL** field to download the root certificate or certificate bundle.
- Step 5** Import the root certificate to the Linux OS on the ECS. For details, see [How Can I Import the Root Certificate to the Windows or Linux OS?](#)

NOTE

- Since April 2017, RDS has offered a new root certificate that has a 20-year validation period. The new certificate takes effect after DB instances are rebooted. Replace the old certificate before it expires to improve system security.
For details, see section [How Can I Identify the Validity Period of an SSL Root Certificate?](#)
- You can also download the certificate bundle, which contains both the new certificate provided since April 2017 and the old certificate.

- Step 6** Connect to an RDS DB instance. The Linux OS is used as an example.

```
mysql -h <hostName> -P <port> -u <userName> -p --ssl-ca=<caName>
```

Table 2-17 Parameter description

Parameter	Description
<hostName>	Indicates the EIP of the DB instance to be connected.
<port>	Indicates the port of the DB instance to be connected.
<userName>	Indicates the username of the RDS database account. The default administrator is root .
<caName>	Indicates the name of the SSL certificate file, which should be stored in the same directory where the command is executed.

For example, to connect to a DB instance through an SSL connection as user **root**, run the following command:

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

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

Enter password:

 **NOTE**

If the connection fails, ensure that preparations have been correctly made in [Preparations](#) and try again.

----End

3 Getting Started with RDS for PostgreSQL

3.1 Connecting to a DB Instance

An RDS DB instance can be connected through a private network or a public network.

Table 3-1 RDS connection methods

Connect Through	IP Address	Scenarios	Description
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.	<ul style="list-style-type: none"> Secure and excellent performance Recommended

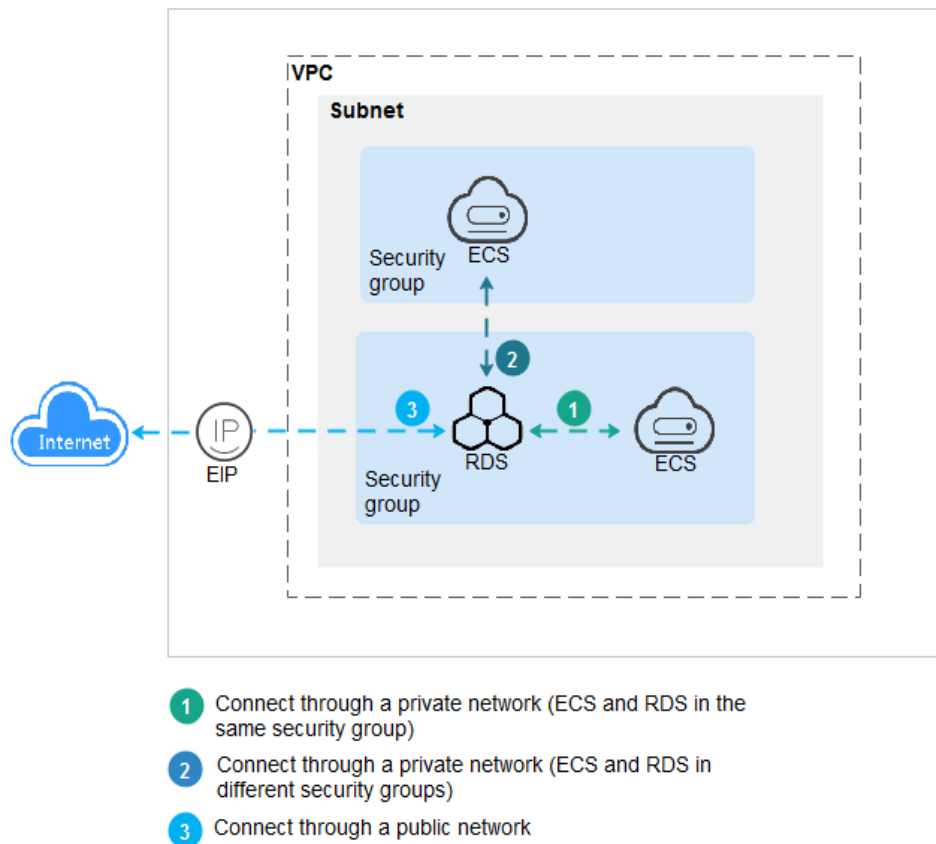
Connect Through	IP Address	Scenarios	Description
Public network	EIP	If you cannot access an RDS DB instance through a floating IP address, bind an EIP to the DB instance and connect the DB instance to the ECS through the EIP.	<ul style="list-style-type: none"> • A relatively lower level of security compared 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 subnet as your RDS DB instance and use a floating IP address to access the DB instance.

 **NOTE**

- VPC: indicates the Virtual Private Cloud.
- ECS: indicates the 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 3-1 illustrates the connection over a private network or a public network.

Figure 3-1 DB instance connection



3.2 Connecting to a PostgreSQL DB Instance Through a Private Network

3.2.1 Overview

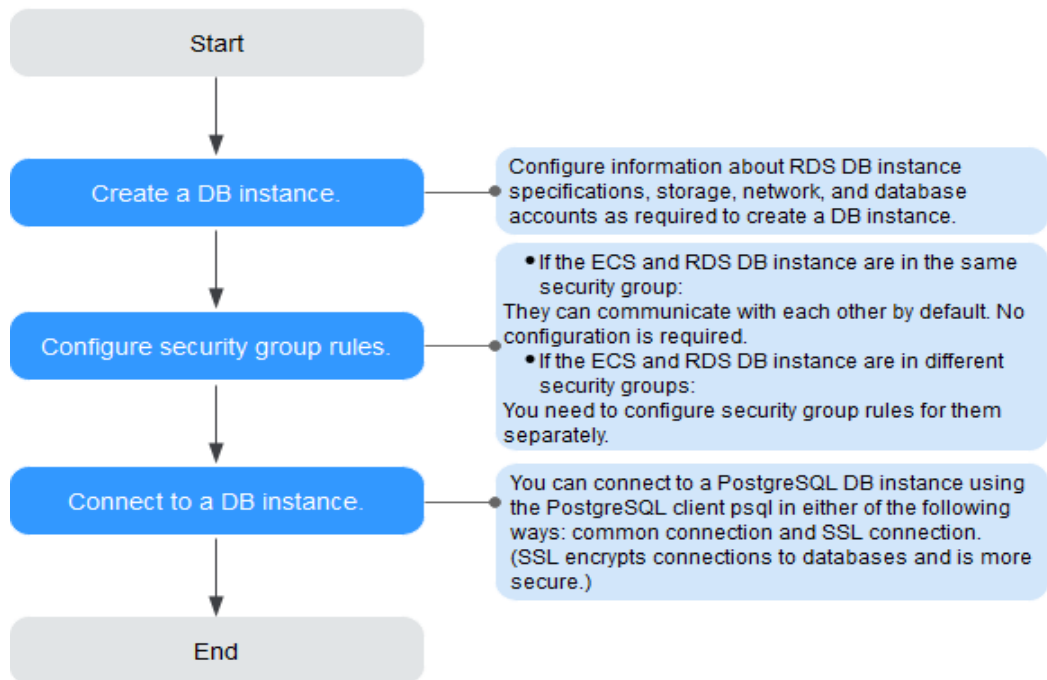
This section describes how to create a DB instance on the management console and connect to the DB instance through an ECS.

If you are using RDS for the first time, see the constraints described in section [PostgreSQL Constraints](#).

Process

Figure 3-2 illustrates the process of connecting to a PostgreSQL DB instance through a private network.

Figure 3-2 Connecting to a DB instance through a private network



- **Step 1: Create a DB instance.** Confirm the specifications, storage, network, and database account configurations of the PostgreSQL DB instances based on service requirements.
- **Step 2: Configure security group rules.**
 - 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 **Step 3: Connect to a DB Instance Through psql.**
 - 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 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.
- **Step 3: Connect to a DB instance through a private network.** You can connect to a DB instance through a common connection or an SSL connection. The SSL connection encrypts data and is more secure. The CLI tool `psql` is used as an example to describe the connection method.

3.2.2 Step 1: Create a DB Instance

Scenarios

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

RDS allows you to tailor your computing resources and storage space to your business needs.

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 **Service List**. Under **Database**, click **Relational Database Service**. The RDS console is displayed.
- Step 4** On the **Instance Management** page, click **Create DB Instance**.
- Step 5** On the displayed page, configure information about your DB instance. Then, click **Create Now**.

Table 3-2 Basic information

Parameter	Description
Region	The region your RDS resources will be located. You can change it on the creation page, or go back to the Instance Management page and change it in the upper left corner. NOTE Products in different regions cannot communicate with each other through a private network and you cannot change the region of a DB instance after creating the instance. Therefore, exercise caution when selecting a region.
DB Instance Name	Must start with a letter and consist of 4 to 64 characters. Only letters, digits, hyphens (-), and underscores (_) are allowed.
DB Engine	Set to PostgreSQL .
DB Engine Version	For details, see DB Engines and Versions . Different DB engine versions are supported in different regions. If you use a PostgreSQL database, select a proper DB engine version based on service requirements. You are advised to select the latest available version because it is more stable, reliable, and secure.

Parameter	Description
DB Instance Type	<ul style="list-style-type: none"> Primary/Standby: uses the 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. An AZ is a physical region where resources use independent power supply and networks. AZs are physically isolated but interconnected through an internal network. RDS allows you to deploy primary/standby DB instances in an AZ or across AZs. You can determine whether the standby AZ is the same as the primary AZ. <ul style="list-style-type: none"> If they are the same (default setting), the primary and standby DB instances are deployed in the same AZ. If they are different, the primary and standby DB instances are deployed in different AZs to ensure failover support and high availability. Single: uses the single-node architecture, which is more cost-effective than mainstream primary/standby DB instances. It is suitable for developing and testing of microsites, and small- and medium-sized enterprises, or for learning about RDS.
Time Zone	Select a time zone when you are creating a DB instance, and you can change it after the DB instance is created.

Table 3-3 Instance specifications

Parameter	Description
Instance Class	<p>Refers to the CPU and memory of a DB instance. Different instance classes have different numbers of database connections and different maximum IOPS.</p> <p>For details about instance classes, see DB Instance Classes.</p> <p>After a DB instance is created, you can change its CPU and memory. For details, see Changing a DB Instance Class.</p>

Parameter	Description
Storage Type	<p>Determines the DB instance read/write speed. The higher the maximum throughput is, the higher the DB instance read/write speed can be.</p> <ul style="list-style-type: none"> • Common I/O: uses the SATA disk type, with a maximum throughput of 90 MB/s. • Ultra-high I/O: supports a maximum throughput of 350 MB/s.
Storage Space (GB)	<p>Contains the file system overhead required for inode, reserved block, and database operation. Can range in size from 40 GB to 4,000 GB and can be scaled up only by a multiple of 10 GB.</p> <p>After a DB instance is created, you can scale up its storage space. For details, see Scaling Up Storage Space.</p>

Table 3-4 Network

Parameter	Description
VPC	<p>A dedicated virtual network in which your RDS DB instances are located. A VPC can isolate networks for different services. You can select an existing VPC or create a VPC. For details on how to create a VPC, see the "Creating a VPC" section in the <i>Virtual Private Cloud User Guide</i>.</p> <p>If no VPC is available, RDS allocates a VPC for you by default.</p> <p>NOTICE After the DB instance is created, the VPC cannot be changed.</p>
Subnet	<p>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.</p> <p>A floating IP address is automatically assigned when you create a DB instance. You can also enter an unused floating IP address in the subnet CIDR block. After the DB instance is created, you can change the floating IP address.</p>
Security Group	<p>Controls the access that traffic has in and out of a DB instance. By default, the security group associated with the DB instance is authorized.</p> <p>Enhances security by controlling access to RDS from other services. You need to add rules to a security group so that you can connect to your DB instance.</p> <p>If no security group is available, RDS allocates a security group to you by default.</p>

Table 3-5 Database configuration

Parameter	Description
Administrator	The default login name for the database is root .
Administrator Password	<p>Must consist of 8 to 32 characters and contain at least three of the following: uppercase letters, lowercase letters, digits, and special characters (~!@#%^*_+?,). Enter a strong password. Periodically change it to improve security and prevent security risks such as brute force cracking.</p> <p>Keep this password secure. The system cannot retrieve it.</p> <p>After a DB instance is created, you can reset this password. For details, see section Resetting the Administrator Password.</p>
Confirm Password	Must be the same as Administrator Password .
Parameter Template	<p>Contains engine configuration values that can be applied to one or more DB instances. If you intend to create primary/standby DB instances, they use the same parameter template. You can modify instance parameters as required after the DB instance is created.</p> <p>NOTICE</p> <p>If you use a custom parameter template when creating a DB instance, the following specification-related parameters in the custom template are not delivered. Instead, the default values are used.</p> <ul style="list-style-type: none"> • maintenance_work_mem • shared_buffers • max_connections • effective_cache_size <p>You can modify the instance parameters as required after the DB instance is created. For details, see Modifying Parameters.</p>

Table 3-6 Tags

Parameter	Description
Tag	Tags an RDS DB instance. This configuration is optional. Adding tags to RDS DB instances helps you better identify and manage the DB instances. A maximum of 10 tags can be added for each DB instance.

Table 3-7 Batch creation

Parameter	Description
Quantity	<p>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.</p> <p>If you create multiple DB instances at a time, they will be named with four digits appended to the DB instance name. For example, if you enter instance, the first instance will be named as instance-0001, the second as instance-0002, and so on.</p>


 **NOTE**

The performance of your DB instance depends on its configurations. Hardware configuration items include the instance specifications, storage type, and storage space.

Step 6 Confirm the specifications.

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

Step 7 To view and manage the DB instance, go to the **Instance Management** page.

- During the creation process, the DB instance status is **Creating**.
- To refresh the DB instance list, click  in the upper right corner of the list. When the creation process is complete, the instance status will change to **Available**.
- The automated backup policy is enabled by default. After the DB instance is created, you can modify the policy as needed. An automated full backup is immediately triggered after a DB instance is created.
- The default database port is **5432**. After a DB instance is created, you can change the database port.

For details, see [Changing the Database Port](#).

----End

3.2.3 Step 2: Configure 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 in a VPC.

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

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 [Step 3: Connect to a DB Instance Through psql](#).
- 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 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.

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 500 security group rules.
- Too many security group rules will increase the first packet latency. You are advised to create a maximum of 50 rules for each security group.
- To access 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

If you use **0.0.0.0/0**, you enable all IP addresses to access RDS DB instances in the security group.

3.2.4 Step 3: Connect to a DB Instance Through psql

You can use the PostgreSQL client `psql` to connect to a DB instance through a common connection or an SSL connection. The SSL connection is encrypted and therefore more secure.

Preparations

1. To connect to a DB instance through an ECS, you need to create an ECS first.
2. Install the PostgreSQL client on the prepared ECS or device.

Common Connection

Step 1 Log in to the ECS or the device that can access RDS.

Step 2 Run the following command to connect to the DB instance:

```
psql --no-readline -U <user> -h <host> -p <port> -d <datastore> -W
```

Table 3-8 Parameter description

Parameter	Description
<user>	Indicates the username of the RDS database account. The default administrator is root .
<host>	Indicates the IP address of the primary DB instance. To obtain this parameter, go to the Basic Information page of the DB instance. If the DB instance is accessed through the ECS, the IP address can be found in the Floating IP Address field in the Connection Information area.
<port>	Indicates the database port in use. The default value is 5432 . To obtain this parameter, go to the Basic Information page of the DB instance. The port number can be found in the Database Port field in the Connection Information area.
<datastore>	Indicates the name of the database (the default database name is postgres).



The parameter **-W** indicates that a password must be entered for the connection. After running this command, you will be prompted to enter a password.

Example:

Run the following command as user **root** to connect to a DB instance:

```
psql --no-readline -U root -h 192.168.0.44 -p 5432 -d postgres -W
----End
```

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 **Service List**. Under **Database**, click **Relational Database Service**. The RDS console is displayed.
- Step 4** In the **DB Information** area on the **Basic Information** page, click  in the **SSL** field to download the root certificate or certificate bundle.
- Step 5** Upload the root certificate to the ECS or save it to the device to be connected to the DB instance.

Import the root certificate to the Linux OS on the ECS. For details, see [How Can I Import the Root Certificate to the Windows or Linux OS?](#)

- Step 6** Connect to an RDS DB instance. The Linux OS is used as an example.

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

Table 3-9 Parameter description

Parameter	Description
<i><host></i>	Indicates the IP address of the primary DB instance. To obtain this parameter, go to the Basic Information page of the DB instance. If the DB instance is accessed through the ECS, the IP address can be found in the Floating IP Address field in the Connection Information area.
<i><port></i>	Indicates the database port in use. The default value is 5432 . To obtain this parameter, go to the Basic Information page of the DB instance. The port number can be found in the Database Port field in the Connection Information area.
<i><database></i>	Indicates the name of the database (the default database name is postgres).
<i><user></i>	Indicates the username of the RDS database account. The default administrator is root .
<i><ca-file-directory></i>	Indicates the directory of the CA certificate for the SSL connection. The certificate should be stored in the directory where the command is executed.
sslmode	Indicates the SSL connection mode. Set it to verify-ca to use a CA to check whether the service is trusted.

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

Password:

For example, to connect to a DB instance through an SSL connection as user **root**, run the following command:

```
psql --no-readline -h 192.168.0.44 -p 5432 "dbname=postgres user=root sslmode=verify-ca sslrootcert=/root/ca.pem"
```

Password:

Step 7 The SSL connection is established if information similar to the following is displayed after you log in to the database:

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

----End

3.3 Connecting to a PostgreSQL DB Instance Through a Public Network

3.3.1 Overview

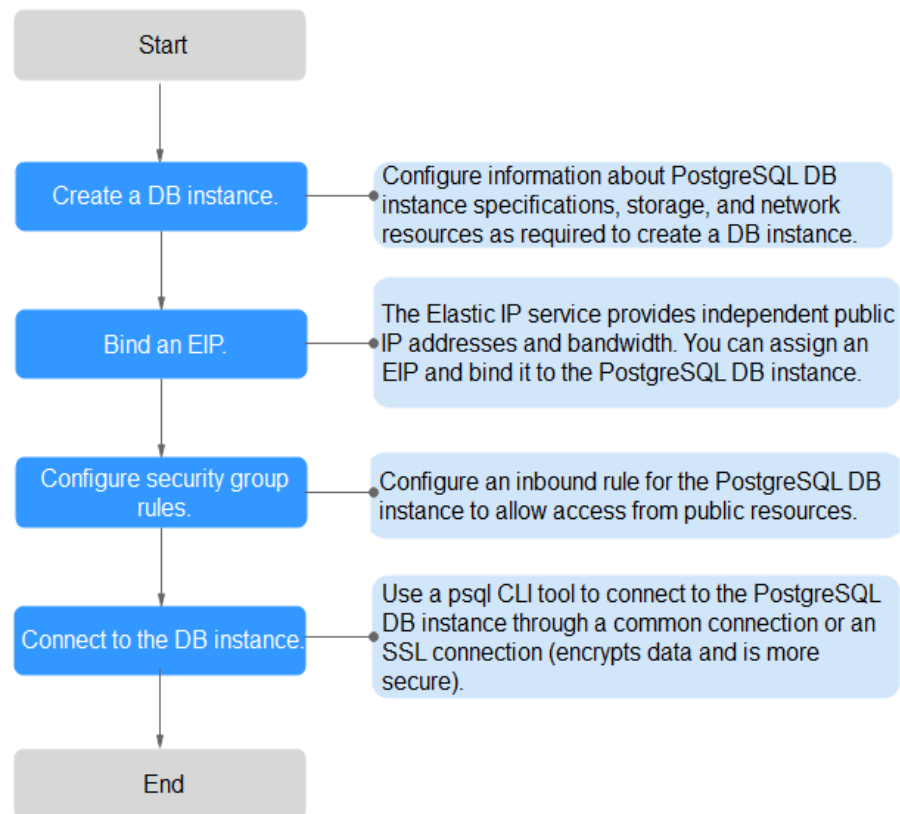
This section describes how to create a PostgreSQL DB instance on the management console and bind an EIP to the DB instance to make the instance publicly accessible.

If you are using RDS for the first time, see the constraints described in section [PostgreSQL Constraints](#).

Process

Figure 3-3 illustrates the process of connecting to a PostgreSQL DB instance through a public network.

Figure 3-3 Connecting to a DB instance through a public network



- **Step 1: Create a DB instance.** Confirm the specifications, storage, network, and database account configurations of the PostgreSQL DB instances based on service requirements.
- **Step 2: Bind an EIP.** The EIP provides independent public IP addresses and bandwidth for Internet access. You can apply for an EIP on the VPC console and bind the EIP to the RDS DB instance.
- **Step 3: Configure security group rules.** To access a DB instance from resources outside the security group, you need to configure an inbound rule for the security group associated with the DB instance.

- **Step 4: Connect to a DB instance through psql.** You can connect to a DB instance through a common connection or an SSL connection. The SSL connection encrypts data and is more secure.

3.3.2 Step 1: Create a DB Instance

Scenarios

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

RDS allows you to tailor your computing resources and storage space to your business needs.

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 **Service List**. Under **Database**, click **Relational Database Service**. The RDS console is displayed.
- Step 4** On the **Instance Management** page, click **Create DB Instance**.
- Step 5** On the displayed page, configure information about your DB instance. Then, click **Create Now**.

Table 3-10 Basic information

Parameter	Description
Region	The region your RDS resources will be located. You can change it on the creation page, or go back to the Instance Management page and change it in the upper left corner. NOTE Products in different regions cannot communicate with each other through a private network and you cannot change the region of a DB instance after creating the instance. Therefore, exercise caution when selecting a region.
DB Instance Name	Must start with a letter and consist of 4 to 64 characters. Only letters, digits, hyphens (-), and underscores (_) are allowed.
DB Engine	Set to PostgreSQL .
DB Engine Version	For details, see DB Engines and Versions . Different DB engine versions are supported in different regions. If you use a PostgreSQL database, select a proper DB engine version based on service requirements. You are advised to select the latest available version because it is more stable, reliable, and secure.

Parameter	Description
DB Instance Type	<ul style="list-style-type: none"> Primary/Standby: uses the 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. An AZ is a physical region where resources use independent power supply and networks. AZs are physically isolated but interconnected through an internal network. RDS allows you to deploy primary/standby DB instances in an AZ or across AZs. You can determine whether the standby AZ is the same as the primary AZ. <ul style="list-style-type: none"> If they are the same (default setting), the primary and standby DB instances are deployed in the same AZ. If they are different, the primary and standby DB instances are deployed in different AZs to ensure failover support and high availability. Single: uses the single-node architecture, which is more cost-effective than mainstream primary/standby DB instances. It is suitable for developing and testing of microsites, and small- and medium-sized enterprises, or for learning about RDS.
Time Zone	Select a time zone when you are creating a DB instance, and you can change it after the DB instance is created.

Table 3-11 Instance specifications

Parameter	Description
Instance Class	<p>Refers to the CPU and memory of a DB instance. Different instance classes have different numbers of database connections and different maximum IOPS.</p> <p>For details about instance classes, see DB Instance Classes.</p> <p>After a DB instance is created, you can change its CPU and memory. For details, see Changing a DB Instance Class.</p>

Parameter	Description
Storage Type	<p>Determines the DB instance read/write speed. The higher the maximum throughput is, the higher the DB instance read/write speed can be.</p> <ul style="list-style-type: none"> • Common I/O: uses the SATA disk type, with a maximum throughput of 90 MB/s. • Ultra-high I/O: supports a maximum throughput of 350 MB/s.
Storage Space (GB)	<p>Contains the file system overhead required for inode, reserved block, and database operation. Can range in size from 40 GB to 4,000 GB and can be scaled up only by a multiple of 10 GB.</p> <p>After a DB instance is created, you can scale up its storage space. For details, see Scaling Up Storage Space.</p>

Table 3-12 Network

Parameter	Description
VPC	<p>A dedicated virtual network in which your RDS DB instances are located. A VPC can isolate networks for different services. You can select an existing VPC or create a VPC. For details on how to create a VPC, see the "Creating a VPC" section in the <i>Virtual Private Cloud User Guide</i>.</p> <p>If no VPC is available, RDS allocates a VPC for you by default.</p> <p>NOTICE After the DB instance is created, the VPC cannot be changed.</p>
Subnet	<p>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.</p> <p>A floating IP address is automatically assigned when you create a DB instance. You can also enter an unused floating IP address in the subnet CIDR block. After the DB instance is created, you can change the floating IP address.</p>
Security Group	<p>Controls the access that traffic has in and out of a DB instance. By default, the security group associated with the DB instance is authorized.</p> <p>Enhances security by controlling access to RDS from other services. You need to add rules to a security group so that you can connect to your DB instance.</p> <p>If no security group is available, RDS allocates a security group to you by default.</p>

Table 3-13 Database configuration

Parameter	Description
Administrator	The default login name for the database is root .
Administrator Password	<p>Must consist of 8 to 32 characters and contain at least three of the following: uppercase letters, lowercase letters, digits, and special characters (~!@#%^*_+?,). Enter a strong password. Periodically change it to improve security and prevent security risks such as brute force cracking.</p> <p>Keep this password secure. The system cannot retrieve it.</p> <p>After a DB instance is created, you can reset this password. For details, see section Resetting the Administrator Password.</p>
Confirm Password	Must be the same as Administrator Password .
Parameter Template	<p>Contains engine configuration values that can be applied to one or more DB instances. If you intend to create primary/standby DB instances, they use the same parameter template. You can modify instance parameters as required after the DB instance is created.</p> <p>NOTICE</p> <p>If you use a custom parameter template when creating a DB instance, the following specification-related parameters in the custom template are not delivered. Instead, the default values are used.</p> <ul style="list-style-type: none"> • maintenance_work_mem • shared_buffers • max_connections • effective_cache_size <p>You can modify the instance parameters as required after the DB instance is created. For details, see Modifying Parameters.</p>

Table 3-14 Tags

Parameter	Description
Tag	Tags an RDS DB instance. This configuration is optional. Adding tags to RDS DB instances helps you better identify and manage the DB instances. A maximum of 10 tags can be added for each DB instance.

Table 3-15 Batch creation

Parameter	Description
Quantity	<p>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.</p> <p>If you create multiple DB instances at a time, they will be named with four digits appended to the DB instance name. For example, if you enter instance, the first instance will be named as instance-0001, the second as instance-0002, and so on.</p>


 **NOTE**

The performance of your DB instance depends on its configurations. Hardware configuration items include the instance specifications, storage type, and storage space.

Step 6 Confirm the specifications.

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

Step 7 To view and manage the DB instance, go to the **Instance Management** page.

- During the creation process, the DB instance status is **Creating**.
- To refresh the DB instance list, click  in the upper right corner of the list. When the creation process is complete, the instance status will change to **Available**.
- The automated backup policy is enabled by default. After the DB instance is created, you can modify the policy as needed. An automated full backup is immediately triggered after a DB instance is created.
- The default database port is **5432**. After a DB instance is created, you can change the database port.

For details, see [Changing the Database Port](#).

----End

3.3.3 Step 2: Bind an EIP

Scenarios

By default, a DB instance is not publicly accessible (not bound with an EIP) after being created. You can bind an EIP to the 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 [Step 3: Configure Security Group Rules](#).

- Public accessibility reduces the security of DB instances. Therefore, exercise caution when deciding to connect to DB instances through a public network. To achieve a higher transmission rate and security level, you are advised to migrate your applications to the ECS that is in the same region as RDS.

Binding an EIP

Step 1 On the **Instance Management** page, click the target DB instance.

Step 2 In the navigation pane on the left, choose **EIPs**. On the displayed page, click **Bind EIP**.

Step 3 In the displayed dialog box, select an EIP and click **OK**.

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

Step 4 On the **EIPs** 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

3.3.4 Step 3: Configure 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 in 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 500 security group rules.
- Too many security group rules will increase the first packet latency. You are advised to create a maximum of 50 rules for each security group.
- To access 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

If you use **0.0.0.0/0**, you enable all IP addresses to access RDS DB instances in the security group.

3.3.5 Step 4: Connect to a DB Instance Through psql

You can use the PostgreSQL client psql to connect to a DB instance through a common connection or an SSL connection. The SSL connection is encrypted and therefore more secure.

Preparations

1. To connect to a DB instance through an EIP, you must:
 - a. Bind the EIP to the DB instance. For details, see section [Step 2: Bind an EIP](#).
 - b. Ensure that the local device can access the EIP.
2. Install the PostgreSQL client on the prepared ECS or device.

Common Connection

Step 1 Log in to the ECS or the device that can access RDS.

Step 2 Run the following command to connect to the DB instance:

```
psql --no-readline -U <user> -h <host> -p <port> -d <datastore> -W
```

Table 3-16 Parameter description

Parameter	Description
<user>	Indicates the username of the RDS database account. The default administrator is root .
<host>	Indicates the IP address of the primary DB instance. To obtain this parameter, go to the Basic Information page of the DB instance. The IP address can be found on the EIPs page.
<port>	Indicates the database port in use. The default value is 5432 . To obtain this parameter, go to the Basic Information page of the DB instance. The port number can be found in the Database Port field in the Connection Information area.
<datastore>	Indicates the name of the database (the default database name is postgres).



The parameter **-W** indicates that a password must be entered for the connection. After running this command, you will be prompted to enter a password.

Example:

Run the following command as user **root** to connect to a DB instance:

```
psql --no-readline -U root -h 192.168.0.44 -p 5432 -d postgres -W
----End
```

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 **Service List**. Under **Database**, click **Relational Database Service**. The RDS console is displayed.
- Step 4** In the **DB Information** area on the **Basic Information** page, click  in the **SSL** field to download the root certificate or certificate bundle.
- Step 5** Upload the root certificate to the ECS or save it to the device to be connected to the DB instance.

Import the root certificate to the Linux OS on the ECS. For details, see [How Can I Import the Root Certificate to the Windows or Linux OS?](#)

- Step 6** Connect to an RDS DB instance. The Linux OS is used as an example.

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

Table 3-17 Parameter description

Parameter	Description
<i><host></i>	Indicates the IP address of the primary DB instance. To obtain this parameter, go to the Basic Information page of the DB instance. The IP address can be found on the EIPs page.
<i><port></i>	Indicates the database port in use. The default value is 5432 . To obtain this parameter, go to the Basic Information page of the DB instance. The port number can be found in the Database Port field in the Connection Information area.
<i><database></i>	Indicates the name of the database (the default database name is postgres).
<i><user></i>	Indicates the username of the RDS database account. The default administrator is root .
<i><ca-file-directory></i>	Indicates the directory of the CA certificate for the SSL connection. The certificate should be stored in the directory where the command is executed.
sslmode	Indicates the SSL connection mode. Set it to verify-ca to use a CA to check whether the service is trusted.

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

Password:

For example, to connect to a DB instance through an SSL connection as user **root**, run the following command:

```
psql --no-readline -h 192.168.0.44 -p 5432 "dbname=postgres user=root  
sslmode=verify-ca sslrootcert=/root/ca.pem"
```

Password:

Step 7 The SSL connection is established if information similar to the following is displayed after you log in to the database:

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

----End

4 Working with RDS for MySQL

4.1 Instance Management

4.1.1 Changing a DB Instance Name

Scenarios


You can change the name of a primary DB instance or read replica.


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 **Service List**. Under **Database**, click **Relational Database Service**. The RDS console is displayed.

Step 4 On the **Instance Management** page, locate the target DB instance and click  next to it to edit the DB instance name. Then, click **OK**.

Alternatively, click the target DB instance to go to the **Basic Information** page. In the **DB Instance Name** field in the **DB Information** area, click  to edit the DB instance name.

The DB instance name must start with a letter and consist of 4 to 64 characters. Only uppercase letters, lowercase letters, digits, hyphens (-), and underscores (_) are allowed.

- To submit the change, click .
- To cancel the change, click .

Step 5 View the result of the change on the **Basic Information** page.

----End


4.1.2 Changing the Failover Priority

Scenarios

You can change the failover priority on availability or reliability to meet service requirements.

- **Reliability:** This option is selected by default. Data consistency is preferentially ensured during the primary/standby failover. This is recommended for users whose highest priority is data consistency.
- **Availability:** Database availability is preferentially ensured during the primary/standby failover. This is recommended for users who require their databases to provide uninterrupted online services.

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 **Service List**. Under **Database**, click **Relational Database Service**. The RDS console is displayed.
 - Step 4** On the **Instance Management** page, click the target primary/standby DB instances.
 - Step 5** In the **DB Information** area on the displayed **Basic Information** page, click **Change** in the **Failover Priority** field. In the displayed dialog box, select a priority and click **OK**.
 - Step 6** View the result of the change on the **Basic Information** page.
- End

4.1.3 Changing a DB Instance Class


Scenarios

You can change the CPU or memory (instance class) of a DB instance as required. If the status of a DB instance changes from **Changing instance class** to **Available**, the change is successful.

NOTE

- A DB instance cannot be deleted while its instance class is being changed.

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 **Service List**. Under **Database**, click **Relational Database Service**. The RDS console is displayed.

Step 4 On the **Instance Management** page, locate the target DB instance and choose **More > Change Instance Class** in the **Operation** column.

Alternatively, click the target DB instance to go to the **Basic Information** page. In the **DB Information** area, click **Change** in the **Instance Class** field.

Step 5 On the displayed page, specify the new instance class and click **Next**.

Step 6 View the DB instance class change result.

- Changing the DB instance class takes 5–15 minutes. During this period, the status of the DB instance on the **Instance Management** page is **Changing instance class**. After a few minutes, click the DB instance and view the instance class on the displayed **Basic Information** page to check that the change is successful.

NOTICE

After you change a MySQL instance class, the values of the following parameters will also be changed accordingly: **back_log**, **innodb_buffer_pool_size**, **innodb_log_buffer_size**, **innodb_log_files_in_group**, **max_connections**, **innodb_page_cleaners**, **innodb_buffer_pool_instances**, **threadpool_size**, and **slave_parallel_workers**.

----End

4.1.4 Scaling Up Storage Space

Scenarios

You can scale up storage space if it is no longer sufficient for your requirements. If the DB instance status is **Storage full** and data cannot be written to databases, you need to scale up storage space.

For details about the causes and solutions of insufficient storage space, see section [What Should I Do If My Data Exceeds the Database Storage Space of an RDS DB Instance?](#)


RDS allows you to scale up storage space of DB instances but you cannot change the storage type. During the scale-up period, services are not interrupted.

NOTE

- DB instances can be scaled up numerous times.
- The DB instance is in **Scaling up** state when its storage space is being scaled up and the backup services are not affected.
- For primary/standby DB instances, scaling up the primary DB instance will cause the standby DB instance to also be scaled up accordingly.
- You cannot reboot or delete a DB instance that is being scaled up.
- Storage space can only be scaled up, not down.

Scaling Up a Primary DB Instance

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 **Service List**. Under **Database**, click **Relational Database Service**. The RDS console is displayed.

Step 4 On the **Instance Management** page, locate the target DB instance and choose **More > Scale Storage Space** in the **Operation** column.

You can also perform the following procedure to scale up storage space:

- Click the target DB instance to enter the **Basic Information** page. In the **Storage Space** area, click **Scale**.

Step 5 On the displayed page, specify the new storage space and click **Next**.

The minimum start value of each scaling is 10 GB. A read replica can be scaled up only by a multiple of 10 GB. The allowed maximum storage space is 4,000 GB.

Step 6 Confirm specifications.

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

Step 7 View the scale-up result.

Scaling up storage space takes 3-5 minutes. During the time period, the status of the DB instance on the **Instance Management** page will be **Scaling up**. Click the DB instance and view the utilization on the displayed **Basic Information** page to verify that the scale-up is successful.

If the DB instance is running the MySQL DB engine, you can view the detailed progress and result of the task on the **Task Center** page. For details, see section [Task Center](#).

----End


Scaling Up a Read Replica

Scaling up the storage space of a read replica does not affect that of the primary DB instance. Therefore, you can separately scale read replicas to meet service requirements. New storage space of read replicas after scaling up must be greater than or equal to that of the primary DB instance.

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 **Service List**. Under **Database**, click **Relational Database Service**. The RDS console is displayed.

Step 4 On the **Instance Management** page, locate the target DB instance and click  in front of it. Locate a read replica to be scaled and choose **More > Scale Storage Space** in the **Operation** column.

You can also perform the following procedure to scale up storage space:

- Click the target DB instance to enter the **Basic Information** page. In the **Storage Space** area, click **Scale**.

Step 5 On the displayed page, specify the new storage space and click **Next**.

The minimum start value of each scaling is 10 GB. A read replica can be scaled up only by a multiple of 10 GB. The allowed maximum storage space is 4,000 GB.

Step 6 Confirm the specifications.

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

Step 7 View the scale-up result.

Scaling up storage space takes 3-5 minutes. During the time period, the status of the read replica on the **Instance Management** page will be **Scaling up**. Click the read replica and view the utilization on the displayed **Basic Information** page to verify that the scale-up is successful.

If the read replica is running the MySQL DB engine, you can view the detailed progress and result of the task on the **Task Center** page. For details, see section [Task Center](#).

----End

4.1.5 Rebooting a DB Instance

Scenarios

You may need to reboot a DB instance during maintenance. For example, after modifying some parameters, you must reboot the DB instance for the modifications to take effect. You can reboot a primary DB instance or a read replica on the management console.



To reboot a DB instance, the following requirements must be met:

- The status of the DB instance is **Available**.
- No backup is being created or no read replica is being created.

NOTICE

- You can reboot a DB instance only when its status is **Available**, **Abnormal**, or **Storage full**. Your database may be unavailable in some cases such as data is being backed up or some modifications are being made.
 - If the DB instance status is **Abnormal**, the reboot may fail.
 - Rebooting a DB instance will cause service interruption. During the reboot process, the DB instance status is **Rebooting**.
 - Rebooting a DB instance will cause the instance unavailability and clear the cached memory in it. To prevent traffic congestion during peak hours, you are advised to reboot the DB instance during off-peak hours.
-

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 **Service List**. Under **Database**, click **Relational Database Service**. The RDS console is displayed.
 - Step 4** On the **Instance Management** page, locate the target DB instance, or click  and then locate the target read replica. Choose **More > Reboot** in the **Operation** column.

Alternatively, click the target DB instance on the **Instance Management** page to go to the **Basic Information** page. In the upper right corner, click **Reboot**.


For primary/standby DB instances, if you reboot the primary DB instance, the standby DB instance is also rebooted automatically.
 - Step 5** In the displayed dialog box, select a scheduled time, and click **Yes**.
 - Step 6** Refresh the DB instance list and view the status of the DB instance. If its status is **Available**, it has rebooted successfully.
- End


4.1.6 Changing a DB Instance Type from Single to Primary/Standby

Scenarios


- RDS enables you to change single DB instances to primary/standby DB instances to improve instance reliability. This operation does not affect the services running on the primary DB instance.
- Primary/standby DB instances support automatic failover. If the primary DB instance fails, the standby DB instance takes over services quickly. You are advised to deploy primary/standby DB instances in different AZs for high availability and disaster recovery.

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 **Service List**. Under **Database**, click **Relational Database Service**. The RDS console is displayed.
- Step 4** On the **Instance Management** page, locate the target DB instance and choose **More > Change Type to Primary/Standby** in the **Operation** column.

Alternatively, click the target DB instance. In the DB instance topology, click  on the left to change type from single to primary/standby.
- Step 5** Select a standby AZ. Other configurations are the same as those of the primary DB instance by default. Confirm the configurations and click **Submit**.

Step 6 After a single DB instance is changed to primary/standby instances, you can view and manage it on the **Instance Management** page.

- The DB instance is in the **Changing type to primary/standby** status. You can view the progress on the **Task Center** page. For details, see section [Task Center](#).
- In the upper right corner of the DB instance list, click  to refresh the list. After the DB instance type is changed to primary/standby, the instance status will change to **Available** and the instance type will change to **Primary/Standby**.

----End

4.1.7 Manually Switching Between Primary and Standby DB Instances

Scenarios

If you choose to create primary/standby DB instances, RDS will create a primary DB instance and a synchronous standby DB instance in the same region. You can access only the primary DB instance. The standby instance serves as a backup. You can manually promote the standby DB instance to the new primary instance for failover support.

Prerequisites

1. A DB instance is running properly.
2. The replication relationship between the primary and standby instances is normal.

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 **Service List**. Under **Database**, click **Relational Database Service**. The RDS console is displayed.

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

Step 5 In the **DB Information** area on the displayed **Basic Information** page, click **Switch** in the **DB Instance Type** field.

Alternatively, click  in the DB instance topology on the **Basic Information** page to perform a primary/standby switchover.


NOTICE

Primary/standby switchover may cause service interruption for some seconds or minutes (determined by the replication delay). If the primary/standby synchronization delay is too long, a small amount of data may get lost. To prevent traffic congestion, you are advised to perform switchover during off-peak hours.

Step 6 In the **Switch Primary/Standby Instances** dialog box, click **Yes** to switch between the primary and standby DB instances.

If the replication status is **Available** and the replication delay is greater than 300s, the primary/standby switchover task cannot be delivered.

Step 7 After a switchover is successful, you can view and manage the DB instance on the **Instance Management** page.

- During the switchover process, the DB instance status is **Switchover in progress**.
- In the upper right corner of the DB instance list, click  to refresh the list. After the switchover is successful, the DB instance status will become **Available**

----End

4.1.8 Selecting Displayed Items

Scenarios


You can customize instance information items displayed on the **Instance Management** page based on your requirements.

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 **Service List**. Under **Database**, click **Relational Database Service**. The RDS console is displayed.

Step 4 On the **Instance Management** page, click  to edit columns displayed in the DB instance list.

- The following items are displayed by default: DB instance name/ID, DB instance type, DB engine version, status, floating IP address, and operation. These displayed default items cannot be customized.
- In a single project, you can select a maximum of 9 items: DB instance name/ID, DB instance type, DB engine version, status, floating IP address, creation time, database port, storage type, and operation.
- In multiple projects, if you have enabled the ProjectMan permissions, you can select a maximum of 9 items: DB instance name/ID, DB instance type, DB

engine version, status, floating IP address, creation time, database port, storage type, and operation.



----End

4.1.9 Exporting DB Instance Information

Scenarios



You can export a DB instance list (containing all or selected DB instances) to view and analyze DB instance information.

Exporting Information About All DB Instances

- 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 **Service List**. Under **Database**, click **Relational Database Service**. The RDS console is displayed.
- Step 4** On the **Instance Management** page, click  in the upper right corner of the page. By default, all DB instances are selected for export. In the displayed dialog box, select the items to be exported and click **Export**.
- Step 5** After the export task is completed, a .csv file is generated locally.

----End

Exporting Information About Selected DB Instances

- 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 **Service List**. Under **Database**, click **Relational Database Service**. The RDS console is displayed.
- Step 4** On the **Instance Management** page, select the target DB instances to be exported and click  in the upper right corner of the page. In the displayed dialog box, select the items to be exported and click **Export**.
- Step 5** After the export task is completed, a .csv file is generated locally.

----End

4.1.10 Creating a Same DB Instance

Scenarios

This section describes how to quickly create the DB instance with the same configurations as the selected one.

 **NOTE**

- You can create DB instances with the same configurations numerous times.
- This function is unavailable for read replicas.

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 **Service List**. Under **Database**, click **Relational Database Service**. The RDS console is displayed.

Step 4 On the **Instance Management** page, locate the target DB instance and choose **More > Create Same DB Instance** in the **Operation** column.

Step 5 On the displayed page, the configurations are the same as those of the selected DB instance. You can change them as required. Then, click **Next**.

For details about MySQL DB instance configurations, see section [Step 1: Create a DB Instance](#).

Step 6 Confirm the specifications.

Step 7 Refresh the DB instance list and view the status of the DB instance. If the status is **Available**, it has been created successfully.

You can manage the DB instance on the **Instance Management** page.

----End

4.1.11 Deleting a DB Instance or Read Replica

Scenarios

You can delete DB instances or read replicas as required on the **Instance Management** page to release resources.

Constraints


DB instance deletion has the following constraints:

- DB instances cannot be deleted when operations are being performed on them. They can be deleted only after the operations are completed.
- If you delete a DB instance, its automated backups are also deleted and you are no longer charged for them. Manual backups are still retained and will incur additional costs.

NOTICE



- If you delete a primary DB instance, its read replicas are also deleted automatically. Exercise caution when performing this operation.
- Deleted DB instances cannot be recovered and resources are released. Exercise caution when performing this operation. If you want to retain data, **create a manual backup** first before deleting the DB instance.

Deleting a DB Instance

- 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 **Service List**. Under **Database**, click **Relational Database Service**. The RDS console is displayed.
- Step 4** On the **Instance Management** page, locate the target primary DB instance to be deleted and click **More > Delete** in the **Operation** column.
- Step 5** In the displayed dialog box, click **Yes**.
- Step 6** Refresh the DB instance list later to check that the deletion is successful.

----End

Deleting a Read Replica

- 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 **Service List**. Under **Database**, click **Relational Database Service**. The RDS console is displayed.
- Step 4** On the **Instance Management** page, locate the target DB instance and click  in front of it. All the read replicas created for the DB instance are displayed.
- Step 5** Locate the target read replicas to be deleted and click **More > Delete** in the **Operation** column.
- Step 6** In the displayed dialog box, click **Yes**.
- Step 7** Refresh the DB instance list later to check that the deletion is successful.

----End

4.2 Read Replicas

4.2.1 Introducing Read Replicas

Introduction

RDS for MySQL supports read replicas.

In read-intensive scenarios, a single DB instance may be unable to handle the read pressure and service performance may be affected. To expand the DB instance read ability to offload read pressure on the database, you can create read replicas in a region. These read replicas can process a large number of read requests and increase application throughput. You need to separately configure connection addresses for the primary DB instance and each read replica on your applications so that all read requests can be sent to read replicas and write requests to the primary DB instance.

A read replica uses the architecture of a single physical node (without a slave node). Changes to the primary DB instance are also automatically synchronized to all associated read replicas through the native replication function of MySQL. The synchronization is not affected by network latency. Read replicas and the primary DB instance must be in the same region but can be in different AZs.

Functions

- Specifications of read replicas can be different from those of the primary DB instance, and can be changed at any time to facilitate flexible scaling.
- You do not need to maintain accounts or databases. Both of them are synchronized from the primary DB instance.
- Read replicas support system performance monitoring.

RDS provides up to 20 monitoring metrics, including storage space, IOPS, number of database connections, CPU usage, and network traffic. You can view these metrics to understand the load of DB instances.

Constraints

- A maximum of five read replicas can be created for a primary DB instance.
- Read replicas do not support backup settings or temporary backups.
- Read replicas do not support using backup files to restore data to the original DB instance or create a DB instance.
- Data cannot be migrated to read replicas.
- Read replicas do not support database creation and deletion.
- Read replicas do not support account creation. You can create accounts only on primary DB instances.

Creating and Managing a Read Replica

- [Creating a Read Replica](#)
- [Managing a Read Replica](#)

4.2.2 Creating a Read Replica

Scenarios

Read replicas are used to enhance the read capabilities and reduce the load on primary DB instances.

After DB instances are created, you can create read replicas for them.


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 **Service List**. Under **Database**, click **Relational Database Service**. The RDS console is displayed.

Step 4 On the **Instance Management** page, locate the target DB instance and click **Create Read Replica** in the **Operation** column.

Alternatively, click the target DB instance. In the DB instance topology, click  under the primary DB instance to create read replicas.

Step 5 On the displayed page, configure information about the DB instance and click **Next**.

Table 4-1 Basic information

Parameter	Description
Region	By default, read replicas are in the same region as the primary DB instance.
DB Instance Name	Must start with a letter and consist of 4 to 64 characters. It can contain only uppercase letters, lowercase letters, digits, hyphens (-), and underscores (_).
DB Engine	Same as the DB engine version of the primary DB instance by default and cannot be changed.
DB Engine Version	Same as the DB engine version of the primary DB instance by default and cannot be changed.
AZ	RDS allows you to deploy primary DB instances and read replicas in a single AZ or across AZs. You can determine whether the read replica AZ is the same as the primary AZ. <ul style="list-style-type: none"> • If they are the same, the read replica and primary DB instance are deployed in the same AZ. • If they are different, the read replica and primary DB instance are deployed in different AZs to ensure data reliability.

Table 4-2 Instance specifications

Parameter	Description
Instance Class	<p>Refers to the CPU and memory of a DB instance. Different instance classes refer to different numbers of database connections and maximum IOPS.</p> <p>For details about instance classes, see section DB Instance Classes.</p> <p>After a DB instance is created, you can change its instance class (CPU and memory). For details, see section Changing a DB Instance Class.</p>
Storage Type	<p>Determines the DB instance read/write speed. The higher the maximum throughput is, the higher the DB instance read/write speed can be.</p> <ul style="list-style-type: none"> • Common I/O: supports a maximum throughput of 90 MB/s. • Ultra-high I/O: supports a maximum throughput of 350 MB/s.
Storage Space	<p>Contains the system overhead required for inode, reserved block, and database operation.</p> <p>By default, storage space of a read replica is the same as that of the primary DB instance.</p>

Table 4-3 Network

Parameter	Description
VPC	Same as the primary DB instance's VPC.
Subnet	Same as the primary DB instance's subnet. A floating IP address is automatically assigned when you create a read replica. You can also enter an unused floating IP address in the subnet CIDR block. After the read replica is created, you can change the floating IP address.
Security Group	Same as the primary DB instance's VPC.

Step 6 Confirm specifications.

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

Step 7 After a read replica is created, you can view and manage it.

For details about how to manage read replicas, see [Managing a Read Replica](#).

You can view the detailed progress and result of the task on the **Task Center** page.

----End


Follow-up Operations

Managing a Read Replica

4.2.3 Managing a Read Replica

Entering the Management Interface Through the Read Replica

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 **Service List**. Under **Database**, click **Relational Database Service**. The RDS console is displayed.

Step 4 In the DB instance list, click  to expand the DB instance details and click the target read replica to go to the **Basic Information** page.

----End

Entering the Management Interface Through the Primary DB Instance

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 **Service List**. Under **Database**, click **Relational Database Service**. The RDS console is displayed.

Step 4 Click the name of the primary DB instance with which the target read replica is associated to go to the **Basic Information** page.

Step 5 In the DB instance topology, click the target read replica. You can view and manage it in the displayed pane.

----End

4.3 Backups and Restorations

4.3.1 Working with Backups

RDS supports backups and restorations to ensure data reliability.

Automated Backups

Automated backups are created during the backup time window of your DB instances. RDS saves automated backups based on the retention period you

specified. If necessary, you can restore to any point in time during your backup retention period.

Manual Backups

Manual backups are user-initiated full backups of DB instances. They are retained until you delete them manually.

4.3.2 Configuring an Automated Backup Policy

Scenarios

When you create a DB instance, an automated backup policy is enabled by default. For security purposes, the automated backup policy cannot be disabled. After the DB instance is created, you can customize the automated backup policy as required and then RDS backs up data based on the automated backup policy you set.

RDS backs up data at the DB instance level, rather than the database level. If a database is faulty or data is damaged, you can restore it from backups to ensure data reliability. Backups are saved as packages in OBS buckets to ensure data confidentiality and durability. Since backing up data affects the database read and write performance, you are advised to set the automated backup time window to off-peak hours.

The automated backup policy is enabled by default as follows:

- Retention period: 7 days
- Time window: An hour within 24 hours, such as 01:00-02:00 or 12:00-13:00. The backup time is configured based on UTC time and is adjusted for daylight saving time, which changes at different times depending on the time zone.
- Backup cycle: Two random days in a week

Modifying an Automated Backup Policy

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 **Service List**. Under **Database**, click **Relational Database Service**. The RDS console is displayed.

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

Step 5 On the **Backups & Restorations** page, click **Modify Backup Policy**.

- **Retention Period** refers to the number of days that your automated backups can be retained. Increasing the retention period will improve data reliability.
- If you shorten the retention period, the new backup policy takes effect for all backup files. The backup files that have expired will be deleted.
- The backup retention period indicates the number of days you want automated full backups and binlog backups of your DB instance to be retained. It ranges from 1–732 days. The backup time window is one hour. You are advised to select an off-peak time window for automated backups. By

default, each day of the week is selected for **Backup Cycle** and you can change it. At least one day must be selected.

Step 6 Click **OK**.

----End

4.3.3 Creating a Manual Backup

Scenarios

RDS allows you to create manual backups of a running primary DB instance. You can use these backups to restore data.

NOTE

When you delete a DB instance, its automated backups are also deleted but its manual backups are retained.

Method 1

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 **Service List**. Under **Database**, click **Relational Database Service**. The RDS console is displayed.

Step 4 On the **Instance Management** page, locate the target DB instance and choose **More > Create Backup** in the **Operation** column.

Step 5 In the displayed dialog box, enter a backup name and description. Then, click **OK**. If you want to cancel the backup creation task, click **Cancel**.

- The backup name must consist of 4 to 64 characters and start with a letter. It can contain only uppercase letters, lowercase letters, digits, hyphens (-), and underscores (_).
- The description consists of a maximum of 256 characters and cannot contain carriage return characters or the following special characters: >!<'&'=
- The time required for creating a manual backup depends on the amount of data.

Step 6 After a manual backup has been created, you can view and manage it on the **Backup Management** page.

Alternatively, click the target DB instance. On the **Backups & Restorations** page, you can view and manage the manual backups.

----End

Method 2

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 **Service List**. Under **Database**, click **Relational Database Service**. The RDS console is displayed.
- Step 4** On the **Instance Management** page, click the target DB instance.
- Step 5** On the **Backups & Restorations** page, click **Create Backup**. In the displayed dialog box, enter a backup name and description and click **OK**. If you want to cancel the backup creation task, click **Cancel**.
- The backup name must consist of 4 to 64 characters and start with a letter. It can contain only uppercase letters, lowercase letters, digits, hyphens (-), and underscores (_).
 - The description consists of a maximum of 256 characters and cannot contain carriage return characters or the following special characters: >!<"&'=
 - The time required for creating a manual backup depends on the amount of data.
- Step 6** After a manual backup has been created, you can view and manage it on the **Backup Management** page.

Alternatively, click the target DB instance. On the **Backups & Restorations** page, you can view and manage the manual backups.

----End

Method 3

- 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 **Service List**. Under **Database**, click **Relational Database Service**. The RDS console is displayed.
- Step 4** On the **Backup Management** page, click **Create Backup**.
- Step 5** In the displayed dialog box, select a primary DB instance, enter a backup name and description, and click **OK**. If you want to cancel the backup creation task, click **Cancel**.
- A backup can only be created for an available primary DB instance. You cannot create a backup for a DB instance that is being backed up or for which a backup is already being created.
 - The backup name must consist of 4 to 64 characters and start with a letter. It can contain only uppercase letters, lowercase letters, digits, hyphens (-), and underscores (_).
 - The description consists of a maximum of 256 characters and cannot contain carriage return characters or the following special characters: >!<"&'=
 - The time required for creating a manual backup depends on the amount of data.
- Step 6** After a manual backup has been created, you can view and manage it on the **Backup Management** page.

Alternatively, click the target DB instance. On the **Backups & Restorations** page, you can view and manage the manual backups.

----End


4.3.4 Downloading a Backup File

Scenarios

This section describes how to download a manual or an automated backup file to a local device and restore data from the backup file.

RDS for MySQL enables you to download full backup files.

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 **Service List**. Under **Database**, click **Relational Database Service**. The RDS console is displayed.
- Step 4** On the **Backup Management** page, locate the target backup to be downloaded and click **Download** in the **Operation** column.

Alternatively, click the target DB instance. In the navigation pane on the left, choose **Backups & Restorations**. On the **Full Backups** page, locate the target backup to be downloaded and click **Download** in the **Operation** column.
- Step 5** In the displayed dialog box, select a method to download backup data.

NOTE

If the size of the backup data is greater than 400 MB, you are advised to use OBS Browser to download the backup data.


- **Use OBS Browser**
 - a. Download OBS Browser
 - b. Decompress and install OBS Browser.
 - c. Log in to OBS Browser.
 - d. Disable certificate verification on OBS Browser.

NOTE

The OBS bucket name displayed in the **Download Backup File** pane on the RDS console does not support certificate verification. Therefore, you need to disable OBS Browser certificate verification before adding the external bucket and then enable it after the backup file is downloaded.

- e. Add an external bucket.
- f. Download the backup file.

On the OBS Browser page, click the bucket that has been successfully added. In the search box on the right of OBS Browser, enter the backup file name displayed on **Download the Backup File** of the RDS console. In the search result, locate the target backup and download it.

- g. After the backup file is downloaded, enable OBS Browser certificate verification.
- **Use Current Browser**
Download the backup file directly from the current browser.
- **Use Download URL**
Click  to copy the URL within the validity period to download backup data. A valid URL for downloading the backup data is displayed.
 - You can use other download tools to download backup data.
 - You can also run the wget command to download backup data.
wget -O FILE_NAME--no-check-certificate "DOWNLOAD_URL"
Variables in the commands are described as follows:
FILE_NAME: indicates the new backup file name after the download. The original backup file name may be too long and exceed the maximum characters allowed by the client file system. You are advised to add **-O** in the wget command to rename the backup file name.
DOWNLOAD_URL: indicates the path of the backup file to be downloaded.

----End


4.3.5 Downloading a Binlog Backup File

Scenarios

This section describes how to download a manual or an automated backup file to a local device and restore data from the backup file.

RDS for MySQL enables you to download binlog backup files.

Downloading a Binlog Backup File

- 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 **Service List**. Under **Database**, click **Relational Database Service**. The RDS console is displayed.
- Step 4** On the **Instance Management** page, click the target DB instance. The **Basic Information** page is displayed.
- Step 5** In the navigation pane on the left, choose **Backups & Restorations**. On the **Binlog Backups** page, locate the target backup to be downloaded and click **Download** in the **Operation** column.

You can also select the binlog backups to be downloaded and click **Download** above the list.
- Step 6** After the download is complete, you can view the binlog backups locally.

----End

4.3.6 Restoring a DB Instance from a Backup

Scenarios

This section describes how to use an automated or manual backup to restore a DB instance to the status when the backup was created. The restoration is at the DB instance level.

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 **Service List**. Under **Database**, click **Relational Database Service**. The RDS console is displayed.

Step 4 On the **Backup Management** page, select the backup to be restored and click **Restore** in the **Operation** column.

Alternatively, click the target DB instance on the **Instance Management** page. On the displayed page, choose **Backups & Restorations**. On the displayed page, select the target backup to be restored and click **Restore** in the **Operation** column.

Step 5 Select a restoration method and click **OK**.

- Create New Instance

The **Create New Instance** page is displayed.

- The DB engine and version are the same as those of the original DB instance and cannot be changed. The database port is **3306** by default and cannot be changed during the restoration.
- Storage space of the new DB instance is the same as that of the original DB instance by default and cannot be less than that of the original DB instance. The administrator password needs to be reset.
- Other settings are the same as those of the original DB instance by default and can be modified. For details, see section [Step 1: Create a DB Instance](#).

- Restore to Original

NOTICE

- If the DB instance for which the backup is created has been deleted, data cannot be restored to the original DB instance.
 - Restoring to the original DB instance will overwrite all existing data and the DB instance will be unavailable during the restoration process.
-

- Restore to Existing

NOTICE

- If the target existing DB instance has been deleted, data cannot be restored to it.
 - Restoring to an existing DB instance will overwrite data on it and cause the existing DB instance to be unavailable.
 - To restore backup data to an existing DB instance, the selected DB instance must use the same DB engine and the same or a later version than the original DB instance.
 - Ensure that the storage space of the selected existing DB instance is greater than or equal to the storage space of the original DB instance. Otherwise, data will not be restored.
-

Select an existing DB instance and click **OK**.

If the automated backup policy is enabled, a full backup will be triggered after the restoration is complete. Otherwise, the full backup will not be triggered.

Step 6 View the restoration result. The result depends on which restoration method was selected:

- **Create New Instance**

A new DB instance is created using the backup data. The status of the DB instance changes from **Creating** to **Available**.

The new DB instance is independent from the original one. If you need read replicas to offload read pressure, create one or more for the new DB instance. After the new instance is created, the system will perform a full backup.

- **Restore to Original**

On the **Instance Management** page, the status of the original DB instance changes from **Restoring** to **Available**. If the original DB instance contains read replicas, the read replica status is the same as the original DB instance status.

If the automated backup policy is enabled, a full backup will be triggered after the restoration is complete. Otherwise, the full backup will not be triggered.

- **Restore to Existing**

On the **Instance Management** page, the status of the target existing DB instance changes from **Restoring** to **Available**. If the target existing DB instance contains read replicas, the read replica status is the same as the target existing DB instance status.

If the automated backup policy is enabled, a full backup will be triggered after the restoration is complete. Otherwise, the full backup will not be triggered.

----End

4.3.7 Restoring a DB Instance to a Point in Time

Scenarios

You can use an automated backup to restore a DB instance to a specified point in time.


RDS for MySQL supports restoration to a new, the original, or an existing DB instance.

Constraints

- If you restore backup data to a new DB instance:
 - The DB engine, version, and port number of the database are the same as those of the original DB instance and cannot be changed.
 - You need to set a new administrator password.

Restoring a DB Instance

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 **Service List**. Under **Database**, click **Relational Database Service**. The RDS console is displayed.

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

Step 5 In the navigation pane on the left, choose **Backups & Restorations**. On the displayed page, click **Restore to Point in Time**.

Step 6 Select a restoration date and time range, enter a time point within the selected time range, and select a restoration method. Then, click **OK**.

- Create New Instance
The **Create New Instance** page is displayed.
 - The DB engine, version, and port number of the database are the same as those of the original DB instance and cannot be changed.
 - You need to set a new administrator password.
 - You can modify the other parameter values.
- Restore to Original

NOTICE

Restoring to the original DB instance will overwrite all existing data and the DB instance will be unavailable during the restoration process.

- Restore to Existing

NOTICE

- Restoring to an existing DB instance will overwrite data on it and cause the existing DB instance to be unavailable.
- To restore backup data to an existing DB instance, the selected DB instance must use the same DB engine and the same or a later version than the original DB instance.
- Ensure that the storage space of the selected DB instance is greater than or equal to the storage space of the original DB instance. Otherwise, data will not be restored.

Select an existing DB instance and click **OK**.

Step 7 View the restoration result. The result depends on which restoration method was selected:

- Create New Instance

A new DB instance is created using the backup data. The status of the DB instance changes from **Creating** to **Available**.

The new DB instance is independent from the original one. If you need read replicas to offload read pressure, create one or more for the new DB instance.

- Restore to Original

On the **Instance Management** page, the status of the DB instance changes from **Restoring** to **Available**.

A new restoration time range is available. There will be a difference between the new and original time ranges. This difference reflects the duration of the restoration.

- Restore to Existing

On the **Instance Management** page, the status of the DB instance changes from **Restoring** to **Available**.

After the restoration, the system will perform a full backup.

----End

4.3.8 Replicating a Backup

Scenarios

This section describes how to replicate a manual or an automated backup. The new backup name must be different from the original backup name.

Constraints

You can replicate backups and use them only in the same region.


Backup Retention Policy

- RDS will delete automated backups when they expire or the DB instance for which the backups are created is deleted.

- If you want to retain the automated backups for a long time, you can replicate them to generate manual backups, which will be always retained unless you delete them.
- If storage space used for manual backups exceeds the default storage space, additional RDS storage costs may incur.

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 **Service List**. Under **Database**, click **Relational Database Service**. The RDS console is displayed.

Step 4 On the **Instance Management** page, click the target DB instance. On the **Backups & Restorations** page, locate the target backup to be replicated and click **Replicate** in the **Operation** column.

Alternatively, choose **Backup Management**. On the displayed page, locate the manual backup to be replicated and choose **More > Replicate** or locate an automated backup and click **Replicate** in the **Operation** column.

Step 5 In the displayed dialog box, enter a new backup name and description and click **OK**.

- The backup name must consist of 4 to 64 characters and start with a letter. It can contain only uppercase letters, lowercase letters, digits, hyphens (-), and underscores (_).
- The description consists of a maximum of 256 characters and cannot contain the following special characters: >!<"&'=

Step 6 After the new backup has been created, you can view and manage it on the **Backup Management** page.

----End

4.3.9 Deleting a Manual Backup

Scenarios


You can delete manual backups to release storage space.

NOTICE

Deleted manual backups cannot be recovered. Exercise caution when performing this operation.

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 **Service List**. Under **Database**, click **Relational Database Service**. The RDS console is displayed.
- Step 4** In the navigation pane on the left, choose **Backup Management**. On the displayed page, locate the target manual backup to be deleted and choose **More > Delete** in the **Operation** column.

The following backups cannot be deleted:

- Automated backups
- Backups that are being restored
- Backups that are being replicated

- Step 5** In the displayed dialog box, click **Yes**.

----End

4.4 Parameter Template Management

4.4.1 Suggestions on Tuning MySQL Parameters

Parameters are key configuration items in a database system. Improper parameter settings may adversely affect the stable running of databases. This section describes some important parameters for your reference. For details, see [MySQL official website](#).

Sensitive Parameters

Example parameters are as follows:

- **lower_case_table_names**

Default value: **1**

Function: Controls whether database and tables stored on disks are case sensitive. If the parameter value is set to **1**, database and table names are lowercase by default. If the parameter value is set to **0**, names stored and name queries are case sensitive.

NOTE

MySQL 8.0 does not support any modifications to this parameter.

Impact: If you change the parameter value of a primary DB instance, you should also change the parameter values for associated read replicas and DB instances restored from the backup. For example, tables **abc** and **Abc** in a primary DB instance is case sensitive, but tables in the associated read replicas and the DB instance restored from the backup are case insensitive. During data synchronization and restoration, errors may occur because the table named **abc** already exists.

- **innodb_flush_log_at_trx_commit**

Default value: **1**

Function: Controls the balance between strict ACID compliance for commit operations and higher performance. The default setting of **1** is required for full ACID compliance. Logs are written and flushed to disks at each transaction commit. If the value is set to **0**, logs are written and flushed to disks once per second. If the value is set to **2**, logs are written at each transaction commit and flushed to disks every two seconds.

Impact: If this parameter is not set to **1**, data security is not guaranteed. When the system fails, data may be lost.

- **sync_binlog**

Default value: **1**

Function: Controls how often the MySQL server synchronizes binary logs to the disk. The default setting of **1** requires synchronization of the binary log to the disk at each transaction commit. If the value is set to **0**, synchronization of the binary log to the disk is not controlled by the MySQL server but relies on the OS to flush the binary log to the disk. This setting provides the best performance, but in the event of a power failure or OS crash, all binary log information in **binlog_cache** is lost.

Impact: If this parameter is not set to **1**, data security is not guaranteed. When the system fails, data may be lost.

Performance Parameters

Relevant parameters are as follows:

- The values of **innodb_spin_wait_delay** and **query_alloc_block_size** are determined by the DB instance specifications. If you increase their values, database performance may be affected.
- If **key_buffer_size** is set to a value smaller than **4096**, the parameter modification will fail.
- If **max_connections** is set to a small value, database access will be affected.
- The default values of the following parameters are determined by the DB instance specifications: **innodb_buffer_pool_size**, **max_connections**, and **back_log**. These parameter values are **default** before being specified.
- The values of **innodb_io_capacity_max** and **innodb_io_capacity** are determined by the storage type. These parameter values are **default** before being specified.

4.4.2 Creating a Parameter Template

You can use database parameter templates to manage the DB engine configuration. A database parameter template acts as a container for engine configuration values that can be applied to one or more DB instances.

This default template contains DB engine defaults and system defaults that are configured based on the engine, compute class, and allocated storage of the instance. You cannot modify the parameter settings of a default parameter template. You must create your own parameter template to change parameter settings.

NOTICE

Not all DB engine parameters can be changed in a custom parameter template.

If you want to use your custom parameter template, you simply create a parameter template and select it when you create a DB instance or apply it to an existing DB instance following the instructions provided in section [Applying a Parameter Template](#).

When you have already created a parameter template and want to include most of the custom parameters and values from that template in a new parameter template, you can replicate that parameter template following the instructions provided in section [Replicating a Parameter Template](#).

The following are the key points you should know when using parameters in a parameter template:

- When you change a dynamic parameter value in a parameter template and save the change, the change takes effect immediately. When you change a static parameter value in a parameter template and save the change, the change will take effect only after you manually reboot the DB instances to which the parameter template applies.
- Improper parameter settings may have unintended adverse effects, including degraded performance and system instability. Exercise caution when modifying database parameters and you need to back up data before modifying parameters in a parameter template. Before applying parameter template changes to a production DB instance, you should try out these changes on a test DB instance.

NOTE

RDS does not share parameter template quotas with DDS.

You can create a maximum of 100 parameter templates for RDS DB instances. All RDS DB engines share the parameter template quota.

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 **Service List**. Under **Database**, click **Relational Database Service**. The RDS console is displayed.

Step 4 On the **Parameter Template Management** page, click **Create Parameter Template**.

Step 5 In the displayed dialog box, configure required information and click **OK**.

- Select a DB engine for the parameter template.
- The template name must consist of 1 to 64 characters. It can contain only uppercase letters, lowercase letters, digits, hyphens (-), underscores (_), and periods (.).

- The description consists of a maximum of 256 characters and cannot contain carriage return characters or the following special characters: >!"&'=

----End

4.4.3 Modifying Parameters

You can modify parameters in a custom parameter template to optimize RDS database performance.

You can change parameter values in custom parameter templates only and cannot change parameter values in default parameter templates.

If you modify a parameter, when the modification takes effect is determined by the type of parameter.

The RDS console displays the statuses of DB instances to which the parameter template applies. For example, if the DB instance has not used the latest modifications made to its parameter template, its status is **Pending reboot**. You need to manually reboot the DB instance for the latest modifications to take effect for that DB instance.

NOTE

RDS has default parameter templates whose parameter values cannot be changed. You can view these parameter values by clicking the default parameter templates. If a custom parameter template is set incorrectly, the database startup may fail. You can re-configure the custom parameter template according to the configurations of the default parameter template.

Modifying Parameter Template Parameters

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 **Service List**. Under **Database**, click **Relational Database Service**. The RDS console is displayed.

Step 4 Choose **Parameter Template Management** in the navigation pane on the left. On the **Custom Templates** page, click the target parameter template.

Step 5 Modify parameters as required.

For detailed parameter description, see section [Suggestions on Tuning MySQL Parameters](#).

Available operations are as follows:

NOTICE

After you modify parameters in a parameter template, some modifications immediately take effect for the DB instance to which the parameter template applies. Exercise caution when performing this operation.

- To save the modifications, click **Save**.
- To cancel the modifications, click **Cancel**.
- To preview the modifications, click **Preview**.

Step 6 After the parameters are modified, you can click **Change History** to view the modification details.

NOTICE

The modifications take effect only after you apply the parameter template to DB instances. For details, see [Applying a Parameter Template](#).

- If you have modified parameters of a primary DB instance, you need to reboot the primary DB instance for the modifications to take effect. (For primary/standby DB instances, the parameter modifications also apply to the standby DB instance.)
- If you have modified parameters of a read replica, you need to reboot the read replica for the modifications to take effect.

----End

Modifying Instance Parameters

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 **Service List**. Under **Database**, click **Relational Database Service**. The RDS console is displayed.

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

Step 5 In the navigation pane on the left, choose **Parameters**. On the displayed page, modify parameters as required.

Available operations are as follows:

NOTICE

After you modify instance parameters, the modifications immediately take effect for the DB instance.

Check the value in the **Effective upon Reboot** column.

- If the value is **Yes** and the DB instance status on the **Instance Management** page is **Pending reboot**, you must reboot the DB instance for the modifications to take effect.
- If the value is **No**, the modifications take effect immediately.

-
- To save the modifications, click **Save**.
 - To cancel the modifications, click **Cancel**.

- To preview the modifications, click **Preview**.

After parameters are modified, you can click **Change History** to view parameter modification details.

----End

4.4.4 Exporting a Parameter Template

Scenarios

- You can export a parameter template of a DB instance for future use. You can apply the exported parameter template to DB instances by referring to section [Applying a Parameter Template](#).
- You can export the parameter template information (parameter names, values, and descriptions) of a DB instance to a CSV file for viewing and analysis.

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 **Service List**. Under **Database**, click **Relational Database Service**. The RDS console is displayed.

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

Step 5 In the navigation pane on the left, choose **Parameters**. On the displayed page, click **Export** above the parameter list.

- Exporting to a custom template

In the displayed dialog box, configure required information and click **OK**.

NOTE

- The template name must consist of 1 to 64 characters. It can contain only uppercase letters, lowercase letters, digits, hyphens (-), underscores (_), and periods (.).
- The description consists of a maximum of 256 characters and cannot contain carriage return characters or the following special characters: >!<'&'=

After the parameter template is exported, a new template is generated in the list on the **Parameter Template Management** page.

- Exporting to a file

The parameter template information (parameter names, values, and descriptions) of a DB instance is exported to a CSV file. In the displayed dialog box, enter the file name and click **OK**.

NOTE

The file name must start with a letter and consist of 4 to 64 characters. It can contain only letters, digits, hyphens (-), and underscores (_).

----End


4.4.5 Comparing Parameter Templates

Scenarios


You can compare DB instance parameters with a parameter template that uses the same DB engine to understand the differences of parameter settings.

You can also compare default parameter templates that use the same DB engine to understand the differences of parameter settings.

Comparing Instance Parameters with a Parameter Template

- 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 **Service List**. Under **Database**, click **Relational Database Service**. The RDS console is displayed.
 - Step 4** On the **Instance Management** page, click the target DB instance.
 - Step 5** In the navigation pane on the left, choose **Parameters**. On the displayed page, click **Compare** above the parameter list.
 - Step 6** In the displayed dialog box, select a parameter template to be compared and click **OK**.
 - If their settings are different, the parameter names and values of both parameter templates are displayed.
 - If their settings are the same, no data is displayed.
- End

Comparing Parameter Templates

- 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 **Service List**. Under **Database**, click **Relational Database Service**. The RDS console is displayed.
 - Step 4** On the **Parameter Template Management** page, click **Custom Templates**. Locate the target parameter template and click **Compare** in the **Operation** column.
 - Step 5** In the displayed dialog box, select a parameter template that uses the same DB engine as the target template and click **OK**.
 - If their settings are different, the parameter names and values of both parameter templates are displayed.
 - If their settings are the same, no data is displayed.
- End

4.4.6 Viewing Parameter Change History


Scenarios

You can view the change history of DB instance parameters or custom parameter templates.

 **NOTE**

An exported or custom parameter template has initially a blank change history.

Viewing Change History of DB Instance Parameters


- 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 **Service List**. Under **Database**, click **Relational Database Service**. The RDS console is displayed.
- Step 4** On the **Instance Management** page, click the target DB instance.
- Step 5** In the navigation pane on the left, choose **Parameters**. On the displayed page, click **Change History**.

You can view the parameter name, original parameter value, new parameter value, modification status, modification time, application status, and application time.

You can apply the parameter template to DB instances as required by referring to section [Applying a Parameter Template](#).

----End

Viewing Change History of a Parameter Template

- 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 **Service List**. Under **Database**, click **Relational Database Service**. The RDS console is displayed.
- Step 4** Choose **Parameter Template Management** in the navigation pane on the left. On the **Custom Templates** page, click the target parameter template.
- Step 5** On the displayed page, choose **Change History** in the navigation pane on the left.

You can view the parameter name, original parameter value, new parameter value, modification status, and modification time.

----End

4.4.7 Replicating a Parameter Template

Scenarios

You can replicate a parameter template you have created. When you have already created a parameter template and want to include most of the custom parameters and values from that template in a new parameter template, you can replicate that parameter template.

After a parameter template is replicated, the new template may be displayed about 5 minutes later.

Default parameter templates cannot be replicated. You can create parameter templates based on the default ones.

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 **Service List**. Under **Database**, click **Relational Database Service**. The RDS console is displayed.

Step 4 On the **Parameter Template Management** page, click **Custom Templates**. Locate the target parameter template and click **Replicate** in the **Operation** column.

Alternatively, click the target DB instance on the **Instance Management** page. On the **Parameters** page, click **Export** to generate a new parameter template for future use.

Step 5 In the displayed dialog box, configure required information and click **OK**.

- The template name must consist of 1 to 64 characters. It can contain only uppercase letters, lowercase letters, digits, hyphens (-), underscores (_), and periods (.).
- The description consists of a maximum of 256 characters and cannot contain carriage return characters or the following special characters: >!<"&'=

After the parameter template is replicated, a new template is generated in the list on the **Parameter Template Management** page.

----End


4.4.8 Resetting a Parameter Template

Scenarios

You can reset all parameters in a custom parameter template to their default settings.

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 **Service List**. Under **Database**, click **Relational Database Service**. The RDS console is displayed.
- Step 4** On the **Parameter Template Management** page, click **Custom Templates**. Locate the target parameter template and choose **More > Reset** in the **Operation** column.
- Step 5** Click **Yes**.

 **NOTE**

After you reset the parameter template, click the DB instance to which the parameter template is applied to view the status of the parameter template. On the displayed **Basic Information** page, if the status of the parameter template is **Pending reboot**, you must reboot the DB instance for the reset to take effect.

----End


4.4.9 Applying a Parameter Template

Scenarios

Modifications to parameters in a custom parameter template take effect only after you apply this parameter template to target DB instances.

- The parameter **innodb_buffer_pool_size** is determined by the memory. DB instances of different specifications have different value ranges. If this parameter value is out of range of the DB instance to which the parameter template applies, the maximum value within the range is used.
- A parameter template can be applied only to DB instances of the same DB engine version.

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 **Service List**. Under **Database**, click **Relational Database Service**. The RDS console is displayed.
- Step 4** On the **Parameter Template Management** page, perform the following operations based on the type of the parameter template to be applied:
- If you intend to apply a default parameter template to DB instances, click **Default Templates**, locate the target parameter template, and click **Apply** in the **Operation** column.
 - If you intend to apply a custom parameter template to DB instances, click **Custom Templates**, locate the target parameter template, and choose **More > Apply** in the **Operation** column.

A parameter template can be applied to one or more DB instances.

- Step 5** In the displayed dialog box, select one or more DB instances to which the parameter template will be applied and click **OK**.

After the parameter template is successfully applied, you can view the application records by referring to section [Viewing Application Records of a Parameter Template](#).

----End

4.4.10 Viewing Application Records of a Parameter Template

Scenarios

You can view the application records of a parameter template.

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 **Service List**. Under **Database**, click **Relational Database Service**. The RDS console is displayed.

Step 4 After a parameter template applies to a DB instance successfully, choose **Parameter Template Management** in the navigation pane on the left, locate the target parameter template, and click **View Application Record** in the **Operation** column on the **Default Templates** page or choose **More > View Application Record** on the **Custom Templates** page.

You can view the name or ID of the DB instance to which the parameter template applies, as well as the application status, application time, and failure cause.

----End

4.4.11 Modifying a Parameter Template Description

Scenarios

You can modify the description of a parameter template you have created.

NOTE


You cannot modify the description of a default parameter template.



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 **Service List**. Under **Database**, click **Relational Database Service**. The RDS console is displayed.

Step 4 On the **Parameter Template Management** page, click **Custom Templates**. Locate the target parameter template and click  in the **Description** column.

- Step 5** Enter a new description. You can click  to submit or  to cancel the modification.
- After you submit the modification, you can view the new description in the **Description** column on the **Parameter Template Management** page.
 - The description consists of a maximum of 256 characters and cannot contain the following special characters: >!<"&'=

----End

4.4.12 Deleting a Parameter Template


Scenarios

You can delete a custom parameter template that is no longer in use.

NOTICE

- Deleted parameter templates cannot be recovered. Exercise caution when performing this operation.
 - Default parameter templates cannot be deleted.
-

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 **Service List**. Under **Database**, click **Relational Database Service**. The RDS console is displayed.
- Step 4** On the **Parameter Template Management** page, click **Custom Templates**. Locate the target parameter template to be deleted and choose **More > Delete** in the **Operation** column.
- Step 5** In the displayed dialog box, click **Yes**.

----End

4.5 Connection Management

4.5.1 Configuring and Changing a Floating IP Address

Scenarios


You can plan and change floating IP addresses after migrating on-premises databases or other cloud databases to RDS.

Configuring a Floating IP Address

You can use a self-assigned IP address when creating a DB instance.

Changing a Floating IP Address

After a DB instance is created, you can change its floating IP address.

- 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 **Service List**. Under **Database**, click **Relational Database Service**. The RDS console is displayed.
- Step 4** On the **Instance Management** page, click the target DB instance.
- Step 5** In the **Connection Information** area on the **Basic Information** page, click **Change** in the **Floating IP Address** field.
- Step 6** In the displayed dialog box, enter a new floating IP address and click **OK**.

----End

4.5.2 Binding and Unbinding an EIP

Scenarios

By default, a DB instance is not publicly accessible (not bound with an EIP) after being created. You can bind an EIP to the DB instance for public accessibility and can unbind the EIP from the DB instance as required.


NOTICE

To ensure that the database can be accessed, the security group used by the database must have the permission to access the database port. For example, if the database port is 8635, ensure that the security group has the permission to access port 8635.

Prerequisites

- You have assigned an EIP on the VPC console.
- You can bind an EIP to a primary DB instance or a read replica only.
- If a DB instance has already been bound with an EIP, you must unbind the EIP from the DB instance first before binding a new EIP to it.


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 **Service List**. Under **Database**, click **Relational Database Service**. The RDS console is displayed.
- Step 4** On the **Instance Management** page, click the target DB instance.

- Step 5** In the navigation pane on the left, choose **Connection Management**. In the **Connection Information** pane on the **Public Connection** page, click **Bind** in the **EIP** field.
- Step 6** In the displayed dialog box, all unbound EIPs are listed. Select the EIP to be bound and click **OK**. If no available EIPs are displayed, click **View EIP** to obtain an EIP.
- Step 7** On the **EIPs** 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.
- To unbind the EIP from the DB instance, see [Unbinding an EIP](#).

----End

Unbinding 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 **Service List**. Under **Database**, click **Relational Database Service**. The RDS console is displayed.
- Step 4** On the **Instance Management** page, click the DB instance that has been bound with an EIP.
- Step 5** In the navigation pane on the left, choose **Connection Management**. In the **Connection Information** pane on the **Public Connection** page, click **Unbind** in the **EIP** field. In the displayed dialog box, click **Yes**.
- Step 6** On the **EIPs** page, view the unbinding result.
- You can also view the progress and result of unbinding an EIP from a DB instance on the **Task Center** page.
- To bind an EIP to the DB instance again, see [Binding an EIP](#).

----End

4.5.3 Changing the Database Port

Scenarios

This section describes how to change the database port of a primary DB instance or a read replica. For primary/standby DB instances, changing the database port of the primary DB instance will cause the database port of the standby DB instance to also be changed accordingly.

Constraints

You cannot perform the following operations when the database port of a DB instance is being changed:

- Bind an EIP to the DB instance.


- Delete the DB instance.
- Create a backup for the DB instance.
- Restore from backup data or from a specific point in time to the original DB instance.


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 **Service List**. Under **Database**, click **Relational Database Service**. The RDS console is displayed.



Step 4 On the **Instance Management** page, click the target DB instance or click  first and then click the target read replica.

Step 5 In the **Connection Information** area on the **Basic Information** page, click  in the **Database Port** field.

You can also choose **Connection Management** in the navigation pane on the left. In the **Connection Information** pane on the **Public Connection** or **Private Connection** page, click  in the **Database Port** field.

NOTE

The MySQL database port ranges from 2100 to 9500.

- To submit the change, click .
 - In the dialog box, click **Yes**.
 - If you change the database port of the primary DB instance, that of the standby DB instance will also be changed and both DB instances will be rebooted.
 - If you change the database port of a read replica, the change will not affect other DB instances. Only the read replica will be rebooted.
 - This process takes 1-5 minutes.
 - In the dialog box, click **No** to cancel the modification.
- To cancel the change, click .

Step 6 View the result of the change on the **Basic Information** page.

----End

4.5.4 Configuring a Security Group Rule

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 in 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.
- 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.
 - 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 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.

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 500 security group rules.
- Too many security group rules will increase the first packet latency. You are advised to create a maximum of 50 rules for each security group.
- To access 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

If you use **0.0.0.0/0**, you enable all IP addresses to access RDS DB instances in the security group.

Procedure

- Step 1** Log in to the management console.
- Step 2** Under **Network**, click **Virtual Private Cloud**.
- Step 3** In the navigation pane on the left, choose **Access Control > Security Groups**.
- Step 4** On the **Security Groups** page, locate the target security group and click **Manage Rule** in the **Operation** column.
- Step 5** On the displayed page, click **Add Rule**.
- Step 6** In the displayed dialog box, set required parameters to add an inbound rule.
- Step 7** Click **OK**.

----End

4.6 Data Migration

4.6.1 Migrating Data to RDS for MySQL Using mysqldump

Preparing for Data Migration

You can access RDS DB instances through an EIP or through an ECS.

1. Prepare an ECS for accessing DB instances in the same VPC or prepare a device for accessing RDS through an EIP.
 - To connect to a DB instance through an ECS, you need to create an ECS first.
For details about how to create and connect to an ECS, see [How Can I Create and Connect to an ECS?](#)
 - To connect to a DB instance through an EIP, you must:
 - i. Bind an EIP to a DB instance. For details, see [Binding an EIP](#).
 - ii. Ensure that the local device can access the EIP.
2. Install the MySQL client on the prepared ECS or device.
For details, see [How Can I Install the MySQL Client?](#)

NOTE

The MySQL client version must be the same as the version of RDS for MySQL. The MySQL database or client will provide mysqldump and mysql.

Exporting Data

Before migrating data to RDS, you need to export data first.

NOTICE

- The export tool must match the DB engine version.
 - Database migration is performed offline. Before the migration, you must stop any applications using the source database.
-

Step 1 Log in to the ECS or the device that can access RDS.

Step 2 Use the mysqldump tool to export metadata into an SQL file.

NOTICE

The MySQL database is required for RDS management. When exporting metadata, do not specify **--all-database**. Otherwise, the MySQL database will be unavailable.

```
mysqldump --databases <DB_NAME> --single-transaction --order-by-primary --hex-blob --no-data --routines --events --set-gtid-purged=OFF -u <DB_USER>
```

```
-p -h <DB_ADDRESS> -P <DB_PORT> |sed -e 's/DEFINER[ ]*=[ ]*[^\]*\*/' -e 's/DEFINER[ ]*=[ ]*.*FUNCTION/FUNCTION/' -e 's/DEFINER[ ]*=[ ]*.*PROCEDURE/PROCEDURE/' -e 's/DEFINER[ ]*=[ ]*.*TRIGGER/TRIGGER/' -e 's/DEFINER[ ]*=[ ]*.*EVENT/EVENT/' > <BACKUP_FILE>
```

- **DB_NAME** indicates the name of the database to be migrated.
- **DB_USER** indicates the database username.
- **DB_ADDRESS** indicates the database address.
- **DB_PORT** indicates the database port.
- **BACKUP_FILE** indicates the name of the file to which the data will be exported.

Enter the database password as prompted.

Example:

```
mysqldump --databases rdsdb --single-transaction --order-by-primary --hex-blob --no-data --routines --events --set-gtid-purged=OFF -u root -p -h 192.168.151.18 -P 3306 |sed -e 's/DEFINER[ ]*=[ ]*[^\]*\*/' -e 's/DEFINER[ ]*=[ ]*.*FUNCTION/FUNCTION/' -e 's/DEFINER[ ]*=[ ]*.*PROCEDURE/PROCEDURE/' -e 's/DEFINER[ ]*=[ ]*.*TRIGGER/TRIGGER/' -e 's/DEFINER[ ]*=[ ]*.*EVENT/EVENT/' > dump-defs.sql
```

Enter password:

NOTE

If you use mysqldump with a version earlier than 5.6, remove **--set-gtid-purged=OFF** before running this command.

After this command is executed, a **dump-defs.sql** file will be generated as follows:

```
[rds@localhost ~]$ ll dump-defs.sql
-rw-r-----. 1 rds rds 2714 Sep 21 08:23 dump-defs.sql
```

Step 3 Use the mysqldump tool to export data into an SQL file.

NOTICE

The MySQL database is required for RDS management. When exporting metadata, do not specify **--all-database**. Otherwise, the MySQL database will be unavailable.

```
mysqldump --databases <DB_NAME> --single-transaction --hex-blob --set-gtid-purged=OFF --no-create-info --skip-triggers -u <DB_USER> -p -h <DB_ADDRESS> -P <DB_PORT> -r <BACKUP_FILE>
```

For details on the parameters in the preceding command, see [Step 2](#).

Enter the database password as prompted.

Example:

```
mysqldump --databases rdsdb --single-transaction --hex-blob --set-gtid-purged=OFF --no-create-info --skip-triggers -u root -p -h 192.168.151.18 -P 8635 -r dump-data.sql
```

NOTE

If you use mysqldump with a version earlier than 5.6, remove `--set-gtid-purged=OFF` before running this command.

After this command is executed, a **dump-data.sql** file will be generated as follows:

```
[rds@localhost ~]$ ll dump-data.sql
-rw-r-----. 1 rds rds 2714 Sep 21 08:23 dump-data.sql
```

----End

Importing Data

You can connect your client to RDS and import exported SQL files into RDS.

NOTICE

If the source database contains triggers, storage processes, functions, or event invocation, you must set **log_bin_trust_function_creators** to **ON** for the destination database before importing data.

Step 1 Import metadata into RDS.

Use the mysql tool to connect to the RDS DB instance, enter the password, and run the following command to import metadata:

```
# mysql -f -h <RDS_ADDRESS> -P <DB_PORT> -u root -p <<BACKUP_DIR>/
dump-defs.sql
```

- **RDS_ADDRESS**: indicates the IP address of the RDS DB instance.
- **DB_PORT**: indicates the RDS DB instance port.
- **BACKUP_DIR**: indicates the directory where **dump-defs.sql** is stored.

Example:

```
# mysql -f -h 172.16.66.198 -P 3306 -u root -p < dump-defs.sql
```

Enter password:

Step 2 Import data into RDS.

```
# mysql -f -h <RDS_ADDRESS> -P <DB_PORT> -u root -p <<BACKUP_DIR>/
dump-data.sql
```

- **RDS_ADDRESS**: indicates the IP address of the RDS DB instance.
- **DB_PORT**: indicates the RDS DB instance port.
- **BACKUP_DIR**: indicates the directory where **dump-data.sql** is stored.

Example:

```
# mysql -f -h 172.16.66.198 -P 3306 -u root -p < dump-data.sql
```

Enter password:

Step 3 View the import result.

```
mysql> show databases;
```

In this example, the database named **my_db** has been imported.

```
mysql> show databases;
+-----+
| Database          |
+-----+
| information_schema |
| my_db             |
| mysql             |
| performance_schema |
+-----+
4 rows in set (0.00 sec)
```

----End

4.7 Data Security

4.7.1 Resetting the Administrator Password

Scenarios

You can reset the administrator password of a primary instance.

You can also reset the password of your database account when using RDS.

You cannot reset the administrator password under the following circumstances:

- The database port is being changed.
- The status of the primary DB instance is **Creating**, **Restoring**, **Rebooting**, **Storage full**, **Changing port**, or **Abnormal**.

Precautions

- If you have changed the administrator password of the primary DB instance, the administrator passwords of the standby DB instance and read replica (if any) will also be changed accordingly.
- The length of time it takes for the new password to take effect depends on the amount of service data currently being processed by the primary DB instance.
- To prevent brute force cracking and ensure system security, change your password periodically.

Method 1

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 **Service List**. Under **Database**, click **Relational Database Service**. The RDS console is displayed.

Step 4 On the **Instance Management** page, locate the target DB instance and choose **More > Reset Password** in the **Operation** column.

Step 5 Enter a new password and confirm the password.

NOTICE

Keep this password secure. The system cannot retrieve it.

The new password must consist of 8 to 32 characters and contain at least three of the following: uppercase letters, lowercase letters, digits, and special characters (~!@#%^*-_+?,). Enter a strong password and periodically change it to improve security, preventing security risks such as brute force cracking.

- To submit the new password, click **OK**.
- To cancel the reset operation, click **Cancel**.

----End

Method 2

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 **Service List**. Under **Database**, click **Relational Database Service**. The RDS console is displayed.

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

Step 5 In the **DB Information** area on the **Basic Information** page, click **Reset Password** in the **Administrator** field.

Step 6 Enter a new password and confirm the password.

NOTICE

Keep this password secure. The system cannot retrieve it.

The new password must consist of 8 to 32 characters and contain at least three of the following: uppercase letters, lowercase letters, digits, and special characters (~!@#%^*-_+?,). Enter a strong password and periodically change it to improve security, preventing security risks such as brute force cracking.

- To submit the new password, click **OK**.
- To cancel the reset operation, click **Cancel**.






----End

4.7.2 Changing a Security Group

Scenarios

This section describes how to change the security group of a primary DB instance or read replica. For primary/standby DB instances, changing the security group of the primary DB instance will cause the security group of the standby DB instance to also be changed accordingly.

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 **Service List**. Under **Database**, click **Relational Database Service**. The RDS console is displayed.
- Step 4** On the **Instance Management** page, click the target DB instance or read replica.
- Step 5** In the **Connection Information** area on the **Basic Information** page, click  in the **Security Group** field.
 - To submit the change, click .
 - To cancel the change, click .
- Step 6** Changing the security group takes 1 to 3 minutes. Click  in the upper right corner on the **Basic Information** page to view the result of the change.

----End

4.8 Metrics and Alarms

4.8.1 Configuring Displayed Metrics

The Agent of an RDS DB instance monitors the metrics and status of the DB instance only and does not collect other data except the monitoring metrics.

Description

This section describes the RDS metrics that can be monitored by Cloud Eye as well as their namespaces and dimensions. You can use APIs provided by Cloud Eye to query the monitoring metrics and alarms generated for RDS.

Namespace

SYS.RDS

DB Instance Monitoring Metrics

[Table 4-4](#) lists the performance metrics of MySQL databases.

Table 4-4 Database performance metrics

Metric	Name	Description	Value Range	Monitored Object	Monitoring Interval (Raw Data)
rds001_cpu_util	CPU Usage	CPU usage of the monitored object	0-100%	Monitored object: ECS Monitored instance type: MySQL instance	1 minute
rds002_mem_util	Memory Usage	Memory usage of the monitored object	0-100%	Monitored object: ECS Monitored instance type: MySQL instance	1 minute
rds003_iops	IOPS	Average number of I/O requests processed by the system in a specified period	≥ 0 counts/s	Monitored object: ECS Monitored instance type: MySQL instance	1 minute
rds004_bytes_in	Network Input Throughput	Incoming traffic in bytes per second	≥ 0 bytes/s	Monitored object: ECS Monitored instance type: MySQL instance	1 minute
rds005_bytes_out	Network Output Throughput	Outgoing traffic in bytes per second	≥ 0 bytes/s	Monitored object: ECS Monitored instance type: MySQL instance	1 minute
rds006_conn_count	Total Connections	Total number of connections that attempt to connect to the MySQL server	≥ 0 counts	Monitored object: database Monitored instance type: MySQL instance	1 minute

Metric	Name	Description	Value Range	Monitored Object	Monitoring Interval (Raw Data)
rds007_conn_active_count	Current Active Connections	Number of current active connections	≥ 0 counts	Monitored object: database Monitored instance type: MySQL instance	1 minute
rds008_qps	QPS	Query times of SQL statements (including stored procedures) per second	≥ 0 queries/s	Monitored object: database Monitored instance type: MySQL instance	1 minute
rds009_tps	TPS	Execution times of submitted and rollback transactions per second	≥ 0 transactions/s	Monitored object: database Monitored instance type: MySQL instance	1 minute
rds010_innodb_buf_usage	Buffer Pool Usage	Ratio of idle pages to the total number of buffer pool pages in the InnoDB buffer	0-1	Monitored object: database Monitored instance type: MySQL instance	1 minute
rds011_innodb_buf_hit	Buffer Pool Hit Ratio	Ratio of read hits to read requests in the InnoDB buffer	0-1	Monitored object: database Monitored instance type: MySQL instance	1 minute
rds012_innodb_buf_dirty	Buffer Pool Dirty Block Ratio	Ratio of dirty data to used pages in the InnoDB buffer	0-1	Monitored object: database Monitored instance type: MySQL instance	1 minute

Metric	Name	Description	Value Range	Monitored Object	Monitoring Interval (Raw Data)
rds013_innodb_reads	InnoDB Read Throughput	Number of read bytes per second in the InnoDB buffer	≥ 0 bytes/s	Monitored object: database Monitored instance type: MySQL instance	1 minute
rds014_innodb_writes	InnoDB Write Throughput	Number of write bytes per second in the InnoDB buffer	≥ 0 bytes/s	Monitored object: database Monitored instance type: MySQL instance	1 minute
rds015_innodb_read_count	InnoDB File Read Frequency	Number of times that InnoDB reads data from files per second	≥ 0 counts/s	Monitored object: database Monitored instance type: MySQL instance	1 minute
rds016_innodb_write_count	InnoDB File Write Frequency	Number of times that InnoDB writes data to files per second	≥ 0 counts/s	Monitored object: database Monitored instance type: MySQL instance	1 minute
rds017_innodb_log_write_req_count	InnoDB Log Write Requests per Second	Number of InnoDB log write requests per second	≥ 0 counts/s	Monitored object: database Monitored instance type: MySQL instance	1 minute
rds018_innodb_log_write_count	InnoDB Log Physical Write Frequency	Number of InnoDB physical write times to log files per second	≥ 0 counts/s	Monitored object: database Monitored instance type: MySQL instance	1 minute

Metric	Name	Description	Value Range	Monitored Object	Monitoring Interval (Raw Data)
rds019_innodb_log_fsync_count	InnoDB Log fsync() Write Frequency	Number of completed fsync() write times to InnoDB log files per second	≥ 0 counts/s	Monitored object: database Monitored instance type: MySQL instance	1 minute
rds020_temp_tbl_count	Temporary Tables	Number of temporary tables automatically created on hard disks when MySQL statements are executed	≥ 0 tables	Monitored object: database Monitored instance type: MySQL instance	1 minute
rds021_myisam_buf_usage	Key Buffer Usage	MyISAM key buffer usage	0-1	Monitored object: database Monitored instance type: MySQL instance	1 minute
rds022_myisam_buf_write_hit	Key Buffer Write Hit Ratio	MyISAM key buffer write hit ratio	0-1	Monitored object: database Monitored instance type: MySQL instance	1 minute
rds023_myisam_buf_read_hit	Key Buffer Read Hit Ratio	MyISAM key buffer read hit ratio	0-1	Monitored object: database Monitored instance type: MySQL instance	1 minute

Metric	Name	Description	Value Range	Monitored Object	Monitoring Interval (Raw Data)
rds024_myisam_disk_write_count	MyISAM Disk Write Frequency	Number of times that indexes are written to disks per second	≥ 0 counts/s	Monitored object: database Monitored instance type: MySQL instance	1 minute
rds025_myisam_disk_read_count	MyISAM Disk Read Frequency	Number of times that indexes are read from disks per second	≥ 0 counts/s	Monitored object: database Monitored instance type: MySQL instance	1 minute
rds026_myisam_buffer_write_count	MyISAM Buffer Pool Write Requests per Second	Number of requests for writing indexes into the MyISAM buffer pool per second	≥ 0 counts/s	Monitored object: database Monitored instance type: MySQL instance	1 minute
rds027_myisam_buffer_read_count	MyISAM Buffer Pool Read Requests per Second	Number of requests for reading indexes from the MyISAM buffer pool per second	≥ 0 counts/s	Monitored object: database Monitored instance type: MySQL instance	1 minute
rds028_comd_ml_delete_count	DELETE Statements per Second	Number of DELETE statements executed per second	≥ 0 queries/s	Monitored object: database Monitored instance type: MySQL instance	1 minute
rds029_comd_ml_insert_count	INSERT Statements per Second	Number of INSERT statements executed per second	≥ 0 queries/s	Monitored object: database Monitored instance type: MySQL instance	1 minute

Metric	Name	Description	Value Range	Monitored Object	Monitoring Interval (Raw Data)
rds030_comd_ml_ins_sel_count	INSERT_SELECT Statements per Second	Number of INSERT_SELECT statements executed per second	≥ 0 queries/s	Monitored object: database Monitored instance type: MySQL instance	1 minute
rds031_comd_ml_rep_count	REPLACE Statements per Second	Number of REPLACE statements executed per second	≥ 0 queries/s	Monitored object: database Monitored instance type: MySQL instance	1 minute
rds032_comd_ml_rep_sel_count	REPLACE_SELECT Statements per Second	Number of REPLACE_SELECT statements executed per second	≥ 0 queries/s	Monitored object: database Monitored instance type: MySQL instance	1 minute
rds033_comd_ml_sel_count	SELECT Statements per Second	Number of SELECT statements executed per second	≥ 0 queries/s	Monitored object: database Monitored instance type: MySQL instance	1 minute
rds034_comd_ml_upd_count	UPDATE Statements per Second	Number of UPDATE statements executed per second	≥ 0 queries/s	Monitored object: database Monitored instance type: MySQL instance	1 minute
rds035_innodb_del_row_count	Row Delete Frequency	Number of rows deleted from the InnoDB table per second	≥ 0 rows/s	Monitored object: database Monitored instance type: MySQL instance	1 minute

Metric	Name	Description	Value Range	Monitored Object	Monitoring Interval (Raw Data)
rds036_innodb_ins_row_count	Row Insert Frequency	Number of rows inserted into the InnoDB table per second	≥ 0 rows/s	Monitored object: database Monitored instance type: MySQL instance	1 minute
rds037_innodb_read_row_count	Row Read Frequency	Number of rows read from the InnoDB table per second	≥ 0 rows/s	Monitored object: database Monitored instance type: MySQL instance	1 minute
rds038_innodb_upd_row_count	Row Update Frequency	Number of rows updated into the InnoDB table per second	≥ 0 rows/s	Monitored object: database Monitored instance type: MySQL instance	1 minute
rds039_disk_util	Storage Space Usage	Storage space usage of the monitored object	0-100%	Monitored object: ECS Monitored instance type: MySQL instance	1 minute
rds047_disk_total_size	Total Storage Space	Total storage space of the monitored object	40-4000 GB	Monitored object: ECS Monitored instance type: MySQL instance	1 minute
rds048_disk_used_size	Used Storage Space	Used storage space of the monitored object	0-4000 GB	Monitored object: ECS Monitored instance type: MySQL instance	1 minute

Metric	Name	Description	Value Range	Monitored Object	Monitoring Interval (Raw Data)
rds049_disk_read_throughput	Disk Read Throughput	Number of bytes read from the disk per second	≥ 0 bytes/s	Monitored object: ECS Monitored instance type: MySQL instance	1 minute
rds050_disk_write_throughput	Disk Write Throughput	Number of bytes written into the disk per second	≥ 0 bytes/s	Monitored object: ECS Monitored instance type: MySQL instance	1 minute
rds051_avg_disk_sec_per_read	Average Time per Disk Read (to be deprecated)	Average time required for each disk read in a specified period	> 0s	Monitored object: ECS Monitored instance type: MySQL instance	1 minute
rds052_avg_disk_sec_per_write	Average Time per Disk Write (to be deprecated)	Average time required for each disk write in a specified period	> 0s	Monitored object: ECS Monitored instance type: MySQL instance	1 minute

Dimension

Key	Value
rds_instance_id	MySQL DB instance ID

4.8.2 Setting Alarm Rules

Scenarios

You can set alarm rules by referring to [Setting Alarm Rules](#) to customize the monitored objects and notification policies and stay aware of the RDS operating status.

The RDS alarm rules include alarm rule names, services, dimensions, monitored objects, metrics, alarm thresholds, monitoring period, and whether to send notifications.

Setting Alarm Rules

- Step 1** Log in to the management console.
- Step 2** Click **Service List** and choose **Management & Deployment > Cloud Eye**.
- Step 3** In the navigation pane on the left, choose **Alarm Management > Alarm Rules**.
- Step 4** On the displayed **Alarm Rules** page, click **Create Alarm Rule**.

----End

4.8.3 Viewing Monitoring Metrics

Scenarios

The Cloud Eye service monitors operating statuses of RDS DB instances. You can view the RDS monitoring metrics on the management console.

Monitored data takes some time for transmission and display. The RDS status displayed on the Cloud Eye console is the status of the last 5 to 10 minutes. If your RDS DB instance is newly created, wait for 5 to 10 minutes and then view the monitoring data.

Prerequisites


- RDS is running properly.
Monitoring metrics of the RDS DB instances that are faulty or have been deleted cannot be displayed on the Cloud Eye console. You can view their monitoring metrics after they are rebooted or restored to be normal.

NOTE

If an RDS DB instance has been faulty for 24 hours, Cloud Eye considers that it does not exist and deletes it from the monitoring object list. You need to manually clear the alarm rules created for the DB instance.

- RDS keeps running properly for about 10 minutes.
For a newly created RDS DB instance, you need to wait for a while before viewing the monitoring metrics.

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 **Service List**. Under **Database**, click **Relational Database Service**. The RDS console is displayed.
- Step 4** On the **Instance Management** page, locate the target DB instance and click **View Metric** in the **Operation** column to go to the Cloud Eye console.

Alternatively, click the target DB instance. On the displayed page, click **View Metric** in the upper right corner of the page to go to the Cloud Eye console.

Step 5 On the Cloud Eye console, view monitoring metrics of the primary DB instance.

Cloud Eye can monitor performance metrics from the last 1 hour, 3 hours, 12 hours, 1 day, 7 days, and 30 days.

----End

4.9 Log Management

4.9.1 Viewing Error Logs

RDS log management allows you to view database-level logs, including error logs and slow SQL query logs.

Error logs contain warning- and error-level logs generated during the database running. These can help you analyze problems with the database.

Viewing Log Details

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 **Service List**. Under **Database**, click **Relational Database Service**. The RDS console is displayed.


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

Step 5 In the navigation pane on the left, choose **Logs**. On the **Error Logs** page, view details about error logs.

- You can select a log level in the upper right corner to view logs of the selected level.

NOTE

For MySQL DB instances, the following levels of logs are displayed:

- ALL
 - ERROR
 - WARNING
 - NOTE
- You can click  in the upper right corner to view error logs generated in different time segments.
 - If the description of a log is truncated, locate the log and move your pointer over the description in the **Description** column to view details.

----End

4.9.2 Viewing Slow Query Logs

Scenarios

Slow query logs record statements that exceed **long_query_time**. You can use these logs to identify and optimize the statements that are executing slowly.

RDS supports the following statement types:

- SELECT
- INSERT
- UPDATE
- DELETE
- CREATE

Viewing Log Details

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 **Service List**. Under **Database**, click **Relational Database Service**. The RDS console is displayed.

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

Step 5 In the navigation pane on the left, choose **Logs**. On the **Slow Query Logs** page, view details about slow query logs.

- You can view the slow query log records of a specified execution statement type or a specific time period.
- The **long_query_time** parameter determines when a slow query log is recorded. However, changes to this parameter do not affect already recorded logs. If **long_query_time** is changed from 1s to 0.1s, none of the previously recorded logs that do not meet the new threshold are deleted. For example, a 1.5s SQL statement that was recorded when the threshold was 1s will not be deleted now that the new threshold is 2s.

----End

4.9.3 Viewing Failover/Switchover Logs

You can view failover or switchover logs of MySQL DB instances to evaluate the impact on services.

NOTE

- Only failover and switchover logs within 30 days are displayed.

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 **Service List**. Under **Database**, click **Relational Database Service**. The RDS console is displayed.

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

Step 5 In the navigation pane on the left, choose **Logs**. On the displayed page, click **Failover/Switchover Logs** to view log details.

----End

4.10 Task Center

4.10.1 Viewing a Task

You can view the progresses and results of tasks on the **Task Center** page.

NOTE

RDS allows you to view and manage the following tasks:

- Creating DB instances
- Rebooting DB instances
- Binding EIPs to DB instances
- Unbinding EIPs from DB instances
- Switching primary/standby DB instances
- Changing single DB instances to primary/standby
- Scaling up storage space
- Creating read replicas
- Restoring data to new DB instances


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 **Service List**. Under **Database**, click **Relational Database Service**. The RDS console is displayed.

Step 4 Choose **Task Center** in the navigation pane on the left. On the displayed page, locate the target task and view the task details.

- To identify the target task, you can use the task name, order ID, or DB instance name/ID, or simply enter the target task name in the search box in the upper right corner.
- You can click  in the upper right corner to view the progress and status of tasks in a specific period. The default period is seven days.
A task can be retained for a maximum of one month.
- You can view the instant tasks in the following statuses:
 - Running
 - Completed

- Failed
----End


4.10.2 Deleting a Task Record

You can delete the task records no longer need to be displayed. The deletion only deletes the task records, and does not delete the DB instances or terminate the tasks that are being executed.

NOTICE

Deleted task records cannot be recovered. Exercise caution when performing this operation.

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 **Service List**. Under **Database**, click **Relational Database Service**. The RDS console is displayed.
- Step 4** Choose **Task Center** in the navigation pane on the left. On the displayed page, locate the target task record to be deleted and click **Delete** in the **Operation** column. In the displayed dialog box, click **Yes**.

You can delete the records of instant tasks in any of the following statuses:

- Running
- Completed
- Failed

----End

5 Working with RDS for PostgreSQL

5.1 Instance Management


5.1.1 Changing a DB Instance Name

Scenarios


You can change the name of a primary DB instance or read replica.


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 **Service List**. Under **Database**, click **Relational Database Service**. The RDS console is displayed.

Step 4 On the **Instance Management** page, locate the target DB instance and click  next to it to edit the DB instance name. Then, click **OK**.

Alternatively, click the target DB instance to go to the **Basic Information** page. In the **DB Instance Name** field in the **DB Information** area, click  to edit the DB instance name.

The DB instance name must start with a letter and consist of 4 to 64 characters. Only uppercase letters, lowercase letters, digits, hyphens (-), and underscores (_) are allowed.

- To submit the change, click .
- To cancel the change, click .

Step 5 View the result of the change on the **Basic Information** page.

----End


5.1.2 Changing the Failover Priority

Scenarios

You can change the failover priority on availability or reliability to meet service requirements.

- **Reliability:** This option is selected by default. Data consistency is preferentially ensured during the primary/standby failover. This is recommended for users whose highest priority is data consistency.
- **Availability:** Database availability is preferentially ensured during the primary/standby failover. This is recommended for users who require their databases to provide uninterrupted online services.

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 **Service List**. Under **Database**, click **Relational Database Service**. The RDS console is displayed.
- Step 4** On the **Instance Management** page, click the target primary/standby DB instances.
- Step 5** In the **DB Information** area on the displayed **Basic Information** page, click **Change** in the **Failover Priority** field. In the displayed dialog box, select a priority and click **OK**.
- Step 6** View the result of the change on the **Basic Information** page.

----End

5.1.3 Changing a DB Instance Class


Scenarios

You can change the CPU or memory (instance class) of a DB instance as required. If the status of a DB instance changes from **Changing instance class** to **Available**, the change is successful.

NOTE

- A DB instance cannot be deleted while its instance class is being changed.
- If the primary DB instance has a read replica, the new DB instance class must be less than or equal to the read replica class. When changing the read replica class, ensure that the selected class is greater than or equal to the current primary instance class.

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 **Service List**. Under **Database**, click **Relational Database Service**. The RDS console is displayed.

Step 4 On the **Instance Management** page, locate the target DB instance and choose **More > Change Instance Class** in the **Operation** column.

Alternatively, click the target DB instance to go to the **Basic Information** page. In the **DB Information** area, click **Change** in the **Instance Class** field.

Step 5 On the displayed page, specify the new instance class and click **Next**.

Step 6 View the DB instance class change result.

Changing the DB instance class takes 5–15 minutes. During this period, the status of the DB instance on the **Instance Management** page is **Changing instance class**. After a few minutes, click the DB instance and view the instance class on the displayed **Basic Information** page to check that the change is successful.

NOTICE

After the CPU or memory of a PostgreSQL DB instance is changed, the system will change the values of the following parameters accordingly:

- **shared_buffers**
- **max_connections**
- **maintenance_work_mem**
- **effective_cache_size**

----End

5.1.4 Scaling Up Storage Space

Scenarios

You can scale up storage space if it is no longer sufficient for your requirements. If the DB instance status is **Storage full** and data cannot be written to databases, you need to scale up storage space.

For details about the causes and solutions of insufficient storage space, see section [What Should I Do If My Data Exceeds the Database Storage Space of an RDS DB Instance?](#)


RDS allows you to scale up storage space of DB instances but you cannot change the storage type. During the scale-up period, services are not interrupted.

 **NOTE**

- DB instances can be scaled up numerous times.
- For primary/standby DB instances, scaling up the primary DB instance will cause the standby DB instance to also be scaled up accordingly.
- You cannot reboot or delete a DB instance that is being scaled up.
- Storage space can only be scaled up, not down.

Scaling Up a Primary DB Instance

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 **Service List**. Under **Database**, click **Relational Database Service**. The RDS console is displayed.

Step 4 On the **Instance Management** page, locate the target DB instance and choose **More > Scale Storage Space** in the **Operation** column.

You can also perform the following procedure to scale up storage space:

- Click the target DB instance to enter the **Basic Information** page. In the **Storage Space** area, click **Scale**.

Step 5 On the displayed page, specify the new storage space and click **Next**.

The minimum start value of each scaling is 10 GB. A DB instance can be scaled up only by a multiple of 10 GB. The allowed maximum storage space is 4,000 GB.

Step 6 Confirm specifications.

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

Step 7 View the scale-up result.

Scaling up storage space takes 3-5 minutes. During the time period, the status of the DB instance on the **Instance Management** page will be **Scaling up**. Click the DB instance and view the utilization on the displayed **Basic Information** page to verify that the scale-up is successful.

----End


Scaling Up a Read Replica

Scaling up the storage space of a read replica does not affect that of the primary DB instance. Therefore, you can separately scale read replicas to meet service requirements. New storage space of read replicas after scaling up must be greater than or equal to that of the primary DB instance.

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 **Service List**. Under **Database**, click **Relational Database Service**. The RDS console is displayed.

Step 4 On the **Instance Management** page, locate the target DB instance and click  in front of it. Locate a read replica to be scaled and choose **More > Scale Storage Space** in the **Operation** column.

You can also perform the following procedure to scale up storage space:

- Click the target DB instance to enter the **Basic Information** page. In the **Storage Space** area, click **Scale**.

Step 5 On the displayed page, specify the new storage space and click **Next**.

The minimum start value of each scaling is 10 GB. A read replica can be scaled up only by a multiple of 10 GB. The allowed maximum storage space is 4,000 GB.

Step 6 Confirm specifications.

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

Step 7 View the scale-up result.

Scaling up storage space takes 3-5 minutes. During this time, the status of the read replica on the **Instance Management** page will be **Scaling up**. Click the read replica and view the utilization on the displayed **Basic Information** page to verify that the scale-up is successful.

----End

5.1.5 Rebooting a DB Instance

Scenarios

You may need to reboot a DB instance during maintenance. For example, after modifying some parameters, you must reboot the DB instance for the modifications to take effect. You can reboot a primary DB instance or a read replica on the management console.

To reboot a DB instance, the following requirements must be met:


- The status of the DB instance is **Available**.
- No backup is being created or no read replica is being created.

NOTICE


- You can reboot a DB instance only when its status is **Available**, **Abnormal**, or **Storage full**. Your database may be unavailable in some cases such as data is being backed up or some modifications are being made.
 - If the DB instance status is **Abnormal**, the reboot may fail.
 - Rebooting a DB instance will cause service interruption. During the reboot process, the DB instance status is **Rebooting**.
 - Rebooting a DB instance will cause the instance unavailability and clear the cached memory in it. To prevent traffic congestion during peak hours, you are advised to reboot the DB instance during off-peak hours.
-

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 **Service List**. Under **Database**, click **Relational Database Service**. The RDS console is displayed.

Step 4 On the **Instance Management** page, locate the target DB instance, or click  and then locate the target read replica. Choose **More > Reboot** in the **Operation** column.

Alternatively, click the target DB instance on the **Instance Management** page to go to the **Basic Information** page. In the upper right corner, click **Reboot**.

For primary/standby DB instances, if you reboot the primary DB instance, the standby DB instance is also rebooted automatically.

Step 5 In the displayed dialog box, select a scheduled time, and click **Yes**.

Step 6 Refresh the DB instance list and view the status of the DB instance. If its status is **Available**, it has rebooted successfully.

----End

5.1.6 Changing a DB Instance Type from Single to Primary/Standby

Scenarios

- RDS enables you to change single DB instances to primary/standby DB instances to improve instance reliability.
- Primary/standby DB instances support automatic failover. If the primary DB instance fails, the standby DB instance takes over services quickly. You are advised to deploy primary/standby DB instances in different AZs for high availability and disaster recovery.
- Anti-affinity deployment is supported for primary/standby DB instances to prevent the entire instance unavailability due to the failure of a single host.


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 **Service List**. Under **Database**, click **Relational Database Service**. The RDS console is displayed.


Step 4 On the **Instance Management** page, locate the target DB instance and choose **More > Change Type to Primary/Standby** in the **Operation** column.

Alternatively, click the target DB instance. In the DB instance topology, click  on the left to change type from single to primary/standby.

Step 5 Select a standby AZ. Other configurations are the same as those of the primary DB instance by default. Confirm the configurations and click **Submit**.

Step 6 After a single DB instance is changed to primary/standby instances, you can view and manage it on the **Instance Management** page.

- The DB instance is in the **Changing type to primary/standby** status. You can view the progress on the **Task Center** page. For details, see [Task Center](#).

- In the upper right corner of the DB instance list, click  to refresh the list. After the DB instance type is changed to primary/standby, the instance status will change to **Available** and the instance type will change to **Primary/Standby**.

----End

5.1.7 Manually Switching Between Primary and Standby DB Instances

Scenarios


If you choose to create primary/standby DB instances, RDS will create a primary DB instance and a synchronous standby DB instance in the same region. You can access only the primary DB instance. The standby instance serves as a backup. You can manually promote the standby DB instance to the new primary instance for failover support.

Prerequisites

1. A DB instance is running properly.
2. The replication relationship between the primary and standby instances is normal.

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 **Service List**. Under **Database**, click **Relational Database Service**. The RDS console is displayed.

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

Step 5 In the **DB Information** area on the displayed **Basic Information** page, click **Switch** in the **DB Instance Type** field.

Alternatively, click  in the DB instance topology on the **Basic Information** page to perform a primary/standby switchover.


NOTICE

Primary/standby switchover may cause service interruption for some seconds or minutes (determined by the replication delay). If the primary/standby synchronization delay is too long, a small amount of data may get lost. To prevent traffic congestion, you are advised to perform switchover during off-peak hours.

Step 6 In the **Switch Primary/Standby Instances** dialog box, click **Yes** to switch between the primary and standby DB instances.

If the replication status is **Available** and the replication delay is greater than 300s, the primary/standby switchover task cannot be delivered.

Step 7 After a switchover is successful, you can view and manage the DB instance on the **Instance Management** page.

- During the switchover process, the DB instance status is **Switchover in progress**.
- In the upper right corner of the DB instance list, click  to refresh the list. After the switchover is successful, the DB instance status will become **Available**

----End

5.1.8 Selecting Displayed Items

Scenarios


You can customize instance information items displayed on the **Instance Management** page based on your requirements.

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 **Service List**. Under **Database**, click **Relational Database Service**. The RDS console is displayed.

Step 4 On the **Instance Management** page, click  to edit columns displayed in the DB instance list.

- The following items are displayed by default: DB instance name/ID, DB instance type, DB engine version, status, floating IP address, and operation. These displayed default items cannot be customized.
- In a single project, you can select a maximum of 9 items: DB instance name/ID, DB instance type, DB engine version, status, floating IP address, creation time, database port, storage type, and operation.
- In multiple projects, if you have enabled the ProjectMan permissions, you can select a maximum of 9 items: DB instance name/ID, DB instance type, DB engine version, status, floating IP address, creation time, database port, storage type, and operation.



----End

5.1.9 Exporting DB Instance Information

Scenarios



You can export a DB instance list (containing all or selected DB instances) to view and analyze DB instance information.

Exporting Information About All DB Instances

- 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 **Service List**. Under **Database**, click **Relational Database Service**. The RDS console is displayed.
- Step 4** On the **Instance Management** page, click  in the upper right corner of the page. By default, all DB instances are selected for export. In the displayed dialog box, select the items to be exported and click **Export**.
- Step 5** After the export task is completed, a .csv file is generated locally.

----End

Exporting Information About Selected DB Instances

- 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 **Service List**. Under **Database**, click **Relational Database Service**. The RDS console is displayed.
- Step 4** On the **Instance Management** page, select the target DB instances to be exported and click  in the upper right corner of the page. In the displayed dialog box, select the items to be exported and click **Export**.
- Step 5** After the export task is completed, a .csv file is generated locally.

----End

5.1.10 Creating a Same DB Instance


Scenarios

This section describes how to quickly create the DB instance with the same configurations as the selected one.

NOTE

- You can create DB instances with the same configurations numerous times.
- This function is unavailable for read replicas.

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 **Service List**. Under **Database**, click **Relational Database Service**. The RDS console is displayed.

- Step 4** On the **Instance Management** page, locate the target DB instance and choose **More > Create Same DB Instance** in the **Operation** column.
- Step 5** On the displayed page, the configurations are the same as those of the selected DB instance. You can change them as required. Then, click **Next**.
- For details about PostgreSQL DB instance configurations, see section [Step 1: Create a DB Instance](#).
- Step 6** Confirm the specifications.
- Step 7** Refresh the DB instance list and view the status of the DB instance. If the status is **Available**, it has been created successfully.

You can manage the DB instance on the **Instance Management** page.

----End

5.1.11 Deleting a DB Instance or Read Replica

Scenarios

You can delete DB instances or read replicas as required on the **Instance Management** page to release resources.

Constraints


DB instance deletion has the following constraints:

- DB instances cannot be deleted when operations are being performed on them. They can be deleted only after the operations are completed.
- If you delete a DB instance, its automated backups are also deleted and you are no longer charged for them. Manual backups are still retained and will incur additional costs.

NOTICE

- If you delete a primary DB instance, its read replicas are also deleted automatically. Exercise caution when performing this operation.
 - Deleted DB instances cannot be recovered and resources are released. Exercise caution when performing this operation. If you want to retain data, [create a manual backup](#) first before deleting the DB instance.
-

Deleting a DB Instance

- 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 **Service List**. Under **Database**, click **Relational Database Service**. The RDS console is displayed.
- Step 4** On the **Instance Management** page, locate the target primary DB instance to be deleted and click **More > Delete** in the **Operation** column.


Step 5 In the displayed dialog box, click **Yes**.

Step 6 Refresh the DB instance list later to check that the deletion is successful.


----End

Deleting a Read Replica

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 **Service List**. Under **Database**, click **Relational Database Service**. The RDS console is displayed.

Step 4 On the **Instance Management** page, locate the target DB instance and click  in front of it. All the read replicas created for the DB instance are displayed.

Step 5 Locate the target read replicas to be deleted and click **More > Delete** in the **Operation** column.

Step 6 In the displayed dialog box, click **Yes**.

Step 7 Refresh the DB instance list later to check that the deletion is successful.

----End

5.2 Read Replicas

5.2.1 Introducing Read Replicas

Introduction

RDS for PostgreSQL supports read replicas.

In read-intensive scenarios, a single DB instance may be unable to handle the read pressure and service performance may be affected. To expand the DB instance read ability to offload read pressure on the database, you can create one or more read replicas in a region. These read replicas can process a large number of read requests and increase application throughput. You need to separately configure connection addresses of the primary DB instance and each read replica on your applications so that all read requests can be sent to read replicas and write requests to the primary DB instance.

A read replica uses the architecture of a single physical node (without a slave node). Changes to the primary DB instance are also automatically synchronized to all associated read replicas through the native replication function of PostgreSQL. The synchronization is not affected by network latency. Read replicas and the primary DB instance must be in the same region but can be in different AZs.

Functions

- Specifications of read replicas can be different from those of the primary DB instance, and can be changed at any time to facilitate flexible scaling.

- You do not need to maintain accounts or databases. Both of them are synchronized from the primary DB instance.
- Read replicas support system performance monitoring. RDS provides up to 20 monitoring metrics, including storage space, IOPS, number of database connections, CPU usage, and network traffic. You can view these metrics to understand the load of DB instances.

Constraints

- A maximum of five read replicas can be created for a primary DB instance.
- Read replicas do not support backup settings or temporary backups.
- Read replicas do not support the creation of temporary DB instances from backup files or point-in-time recovery, and do not support overwriting of DB instances from backup files.
- Data cannot be migrated to read replicas.
- Read replicas do not support database creation and deletion.
- Read replicas do not support account creation. You can create accounts only on primary DB instances.
- The specifications of read replicas must be greater than or equal to the specifications of the current primary DB instance.

Creating and Managing a Read Replica

- [Creating a Read Replica](#)
- [Managing a Read Replica](#)

5.2.2 Creating a Read Replica

Scenarios

Read replicas are used to enhance the read capabilities of primary DB instances and reduce the load on primary DB instances.


After DB instances are created, you can create read replicas for them.


NOTE

A maximum of five read replicas can be created for a primary DB instance.

The specifications of read replicas must be greater than or equal to the specifications of the current primary DB instance.

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 **Service List**. Under **Database**, click **Relational Database Service**. The RDS console is displayed.
- Step 4** On the **Instance Management** page, locate the target DB instance and click **Create Read Replica** in the **Operation** column.

Alternatively, click the target DB instance. In the DB instance topology, click  under the primary DB instance to create read replicas.

Step 5 On the displayed page, configure information about the DB instance and click **Next**.

Table 5-1 Basic information

Parameter	Description
Region	By default, read replicas are in the same region as the primary DB instance.
DB Instance Name	Must start with a letter and consist of 4 to 64 characters. It can contain only uppercase letters, lowercase letters, digits, hyphens (-), and underscores (_).
DB Engine	Same as the DB engine version of the primary DB instance by default and cannot be changed.
DB Engine Version	Same as the DB engine version of the primary DB instance by default and cannot be changed.
AZ	RDS allows you to deploy primary DB instances and read replicas in a single AZ or across AZs. You can determine whether the read replica AZ is the same as the primary AZ. <ul style="list-style-type: none"> • If they are the same, the read replica and primary DB instance are deployed in the same AZ. • If they are different, the read replica and primary DB instance are deployed in different AZs to ensure data reliability.

Table 5-2 Instance specifications

Parameter	Description
Instance Class	Refers to the CPU and memory of a DB instance. Different instance classes refer to different numbers of database connections and maximum IOPS.
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. <ul style="list-style-type: none"> • Common I/O: uses the SATA disk type that supports a maximum throughput of 90 MB/s. • Ultra-high I/O: supports a maximum throughput of 350 MB/s.


Parameter	Description
Storage Space	Contains the system overhead required for inode, reserved block, and database operation. By default, storage space of a read replica is the same as that of the primary DB instance.

Table 5-3 Network

Parameter	Description
VPC	Same as the primary DB instance's VPC.
Subnet	Same as the primary DB instance's subnet. A floating IP address is automatically assigned when you create a read replica. You can also enter an unused floating IP address in the subnet CIDR block. After the read replica is created, you can change the floating IP address.
Security Group	Same as the primary DB instance's VPC.

Step 6 Confirm specifications.

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

Step 7 After a read replica has been created, you can view and manage it on the **Instance Management** page by clicking  on the left of the DB instance to which it belongs.

Alternatively, click the target DB instance. In the DB instance topology, click the target read replica. You can view and manage it in the displayed pane.

----End

Follow-up Operations

Managing a Read Replica


5.2.3 Managing a Read Replica

Entering the Management Interface Through the Read Replica

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 **Service List**. Under **Database**, click **Relational Database Service**. The RDS console is displayed.

Step 4 In the DB instance list, click  to expand the DB instance details and click the target read replica to go to the **Basic Information** page.

----End

Entering the Management Interface Through the Primary DB Instance

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 **Service List**. Under **Database**, click **Relational Database Service**. The RDS console is displayed.

Step 4 Click the name of the primary DB instance with which the target read replica is associated to go to the **Basic Information** page.

Step 5 In the DB instance topology, click the target read replica. You can view and manage it in the displayed pane.

----End

5.3 Backups and Restorations

5.3.1 Working with Backups

RDS supports backups and restorations to ensure data reliability.

Automated Backups

Automated backups are created during the backup time window of your DB instances. RDS saves automated backups based on the retention period you specified. If necessary, you can restore to any point in time during your backup retention period.

Manual Backups

Manual backups are user-initiated full backups of DB instances. They are retained until you delete them manually.

5.3.2 Configuring an Automated Backup Policy

Scenarios

When you create a DB instance, an automated backup policy is enabled by default. For security purposes, the automated backup policy cannot be disabled. After the DB instance is created, you can customize the automated backup policy as required and then RDS backs up data based on the automated backup policy you set.

RDS backs up data at the DB instance level, rather than the database level. If a database is faulty or data is damaged, you can restore it from backups to ensure data reliability. Backups are saved as packages in OBS buckets to ensure data

confidentiality and durability. Since backing up data affects the database read and write performance, you are advised to set the automated backup time window to off-peak hours.

The automated backup policy is enabled by default as follows:

- Retention period: 7 days
- Time window: An hour within 24 hours, such as 01:00-02:00 or 12:00-13:00. The backup time is configured based on UTC time and is adjusted for daylight saving time, which changes at different times depending on the time zone.
- Backup cycle: Each day of the week

Modifying an Automated Backup Policy

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 **Service List**. Under **Database**, click **Relational Database Service**. The RDS console is displayed.

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

Step 5 On the **Backups & Restorations** page, click **Modify Backup Policy**.

- **Retention Period** refers to the number of days that your automated backups can be retained. Increasing the retention period will improve data reliability.
- If you shorten the retention period, the new backup policy takes effect for all backup files. The backup files that have expired will be deleted.
- The backup retention period indicates the number of days you want automated full and incremental backups of your DB instance to be retained. It ranges from 1–732 days. The backup time window is one hour. You are advised to select an off-peak time window for automated backups. By default, each day of the week is selected for **Backup Cycle** and you can change it. At least one day must be selected.

Step 6 Click **OK**.

----End

5.3.3 Creating a Manual Backup

Scenarios


RDS allows you to create manual backups of a running primary DB instance. You can use these backups to restore data.

NOTE

When you delete a DB instance, its automated backups are also deleted but its manual backups are retained.

Method 1


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 **Service List**. Under **Database**, click **Relational Database Service**. The RDS console is displayed.
- Step 4** On the **Instance Management** page, locate the target DB instance and choose **More > Create Backup** in the **Operation** column.
- Step 5** In the displayed dialog box, enter a backup name and description. Then, click **OK**. If you want to cancel the backup creation task, click **Cancel**.
- The backup name must consist of 4 to 64 characters and start with a letter. It can contain only uppercase letters, lowercase letters, digits, hyphens (-), and underscores (_).
 - The description consists of a maximum of 256 characters and cannot contain carriage return characters or the following special characters: >!<"&'=
 - The time required for creating a manual backup depends on the amount of data.
- Step 6** After a manual backup has been created, you can view and manage it on the **Backup Management** page.

Alternatively, click the target DB instance. On the **Backups & Restorations** page, you can view and manage the manual backups.

----End


Method 2

- 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 **Service List**. Under **Database**, click **Relational Database Service**. The RDS console is displayed.
- Step 4** On the **Instance Management** page, click the target DB instance.
- Step 5** On the **Backups & Restorations** page, click **Create Backup**. In the displayed dialog box, enter a backup name and description and click **OK**. If you want to cancel the backup creation task, click **Cancel**.
- The backup name must consist of 4 to 64 characters and start with a letter. It can contain only uppercase letters, lowercase letters, digits, hyphens (-), and underscores (_).
 - The description consists of a maximum of 256 characters and cannot contain carriage return characters or the following special characters: >!<"&'=
 - The time required for creating a manual backup depends on the amount of data.
- Step 6** After a manual backup has been created, you can view and manage it on the **Backup Management** page.

Alternatively, click the target DB instance. On the **Backups & Restorations** page, you can view and manage the manual backups.

----End

Method 3

- 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 **Service List**. Under **Database**, click **Relational Database Service**. The RDS console is displayed.
- Step 4** On the **Backup Management** page, click **Create Backup**.
- Step 5** In the displayed dialog box, select a primary DB instance, enter a backup name and description, and click **OK**. If you want to cancel the backup creation task, click **Cancel**.
- A backup can only be created for an available primary DB instance. You cannot create a backup for a DB instance that is being backed up or for which a backup is already being created.
 - The backup name must consist of 4 to 64 characters and start with a letter. It can contain only uppercase letters, lowercase letters, digits, hyphens (-), and underscores (_).
 - The description consists of a maximum of 256 characters and cannot contain carriage return characters or the following special characters: >!<"&'=
 - The time required for creating a manual backup depends on the amount of data.
- Step 6** After a manual backup has been created, you can view and manage it on the **Backup Management** page.

Alternatively, click the target DB instance. On the **Backups & Restorations** page, you can view and manage the manual backups.

----End

5.3.4 Restoring a DB Instance to a Point in Time

Scenarios

You can use an automated backup to restore a DB instance to a specified point in time.


RDS for PostgreSQL supports restoration to a new, the original, or an existing DB instance.

Constraints

- If you restore backup data to a new DB instance:
 - The DB engine, version, and port number of the database are the same as those of the original DB instance and cannot be changed.
 - You need to set a new administrator password.

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 **Service List**. Under **Database**, click **Relational Database Service**. The RDS console is displayed.
- Step 4** On the **Instance Management** page, click the target DB instance.
- Step 5** In the navigation pane on the left, choose **Backups & Restorations**. On the displayed page, click **Restore to Point in Time**.
- Step 6** Select a restoration date and time range, enter a time point within the selected time range, and select a restoration method. Then, click **OK**.

- Create New Instance

The **Create New Instance** page is displayed.

- The DB engine and version of the new DB instance are the same as those of the original DB instance and cannot be changed.
- Storage space of the new DB instance is the same as that of the original DB instance by default and cannot be less than that of the original DB instance. The administrator password needs to be reset.
- Other settings are the same as those of the original DB instance by default and can be modified. For details, see section [Step 1: Create a DB Instance](#).

- Restore to Original

NOTICE

Restoring to the original DB instance will overwrite all existing data and the DB instance will be unavailable during the restoration process.

- Restore to Existing

NOTICE

- Restoring to an existing DB instance will overwrite data on it and cause the existing DB instance to be unavailable.
- To restore backup data to an existing DB instance, the selected DB instance must be in the same VPC as the original DB instance and must have the same DB engine and the same or later version than the original DB instance.
- Ensure that the storage space of the selected DB instance is greater than or equal to the storage space of the original DB instance. Otherwise, data will not be restored.

Select an existing DB instance and click **OK**.

- Step 7** View the restoration result. The result depends on which restoration method was selected:

- Create New Instance

A new DB instance is created using the backup data. The status of the DB instance changes from **Creating** to **Available**.

The new DB instance is independent from the original one. If you need read replicas to offload read pressure, create one or more for the new DB instance.

- Restore to Original

On the **Instance Management** page, the status of the DB instance changes from **Restoring** to **Available**.

A new restoration time range is available. There will be a difference between the new and original time ranges. This difference reflects the duration of the restoration.

- Restore to Existing

On the **Instance Management** page, the status of the DB instance changes from **Restoring** to **Available**.

After the restoration, the system will perform a full backup.

----End


5.3.5 Restoring a DB Instance from a Backup

Scenarios

This section describes how to use an automated or manual backup to restore a DB instance to the status when the backup was created. The restoration is at the DB instance level.

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 **Service List**. Under **Database**, click **Relational Database Service**. The RDS console is displayed.

Step 4 On the **Backup Management** page, select the backup to be restored and click **Restore** in the **Operation** column.

Alternatively, click the target DB instance on the **Instance Management** page. On the displayed page, choose **Backups & Restorations**. On the displayed page, select the target backup to be restored and click **Restore** in the **Operation** column.

Step 5 Select a restoration method and click **OK**.

- Create New Instance

The **Create New Instance** page is displayed.

- The DB engine and version are the same as those of the original DB instance and cannot be changed. The database port is **5432** by default and cannot be changed during the restoration.
- Storage space of the new DB instance is the same as that of the original DB instance by default and cannot be less than that of the original DB instance. The administrator password needs to be reset.

- Restore to Original

NOTICE

- If the DB instance for which the backup is created has been deleted, data cannot be restored to the original DB instance.
- Restoring to the original DB instance will overwrite all existing data and the DB instance will be unavailable during the restoration process.

-
- Restore to Existing

NOTICE

- If the target existing DB instance has been deleted, data cannot be restored to it.
- Restoring to an existing DB instance will overwrite data on it and cause the existing DB instance to be unavailable.
- To restore backup data to an existing DB instance, the selected DB instance must be in the same VPC as the original DB instance and must have the same DB engine and the same or later version than the original DB instance.
- Ensure that the storage space of the selected DB instance is greater than or equal to the storage space of the original DB instance. Otherwise, data will not be restored.

Select an existing DB instance and click **OK**.

If the automated backup policy is enabled, a full backup will be triggered after the restoration is complete. Otherwise, the full backup will not be triggered.

Step 6 View the restoration result. The result depends on which restoration method was selected:

- Create New Instance

A new DB instance is created using the backup data. The status of the DB instance changes from **Creating** to **Available**.

The new DB instance is independent from the original one. If you need read replicas to offload read pressure, create one or more for the new DB instance.

After the new instance is created, the system will perform a full backup.

- Restore to Original

On the **Instance Management** page, the status of the target existing DB instance changes from **Restoring** to **Available**. If the original existing DB instance contains read replicas, the read replica status is the same as the original DB instance status.

If the automated backup policy is enabled, a full backup will be triggered after the restoration is complete. Otherwise, the full backup will not be triggered.

- Restore to Existing

On the **Instance Management** page, the status of the target existing DB instance changes from **Restoring** to **Available**. If the target existing DB instance contains read replicas, the read replica status is the same as the target existing DB instance status.

If the automated backup policy is enabled, a full backup will be triggered after the restoration is complete. Otherwise, the full backup will not be triggered.

----End

5.3.6 Downloading a Backup File

Scenarios

This section describes how to download a manual or an automated backup file to a local device and restore data from the backup file.

You can download both full and incremental backup files of PostgreSQL DB instances.

Downloading Full Backup

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 **Service List**. Under **Database**, click **Relational Database Service**. The RDS console is displayed.

Step 4 On the **Backup Management** page, locate the target backup to be downloaded and click **Download** in the **Operation** column.

Alternatively, click the target DB instance. In the navigation pane on the left, choose **Backups & Restorations**. On the **Full Backups** page, locate the target backup to be downloaded and click **Download** in the **Operation** column.

Step 5 In the displayed dialog box, select a method to download backup data.

NOTE

If the size of the backup data is greater than 400 MB, you are advised to use OBS Browser to download the backup data.

- **Use OBS Browser**
 - a. Download OBS Browser.
 - b. Decompress and install OBS Browser.
 - c. Log in to OBS Browser.
 - d. Disable certificate verification on OBS Browser.

NOTE

The OBS bucket name displayed in the **Download Backup File** pane on the RDS console does not support certificate verification. Therefore, you need to disable OBS Browser certificate verification before adding the external bucket and then enable it after the backup file is downloaded.

- e. Add an external bucket.
- f. Download the backup file.


On the OBS Browser page, click the bucket that has been successfully added. In the search box on the right of OBS Browser, enter the backup file name displayed on **Download the Backup File** of the RDS console. In the search result, locate the target backup and download it.

- g. After the backup file is downloaded, enable OBS Browser certificate verification.

- **Use Current Browser**

Download the backup file directly from the current browser.

- **Use Download URL**

Click  to copy the URL within the validity period to download backup data. A valid URL for downloading the backup data is displayed.

- You can use other download tools to download backup data.
- You can also run the wget command to download backup data.

wget -O FILE_NAME--no-check-certificate " DOWNLOAD_URL"

Variables in the commands are described as follows:

FILE_NAME: indicates the new backup file name after the download is successful. The original backup file name may be too long and exceed the maximum characters allowed by the client file system. You are advised to add **-O** in the wget command to rename the backup file name.

DOWNLOAD_URL: indicates the path of the backup file to be downloaded.

Step 6 Restore data locally as required.

----End

Downloading Incremental Backup

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 **Service List**. Under **Database**, click **Relational Database Service**. The RDS console is displayed.

Step 4 On the **Instance Management** page, click the target DB instance. Choose **Backups & Restorations** in the navigation pane on the left. On the **Incremental Backups** page, locate the target backup to be downloaded and click **Download** in the **Operation** column.

You can also select the incremental backups to be downloaded and click **Download** above the list.

Step 5 After the download is complete, you can view the incremental backups locally.

----End

5.3.7 Replicating a Backup

Scenarios

This section describes how to replicate a manual or an automated backup. The new backup name must be different from the original backup name.

Constraints


You can replicate backups and use them only in the same region.

Backup Retention Policy

- RDS will delete automated backups when they expire or the DB instance for which the backups are created is deleted.
- If you want to retain the automated backups for a long time, you can replicate them to generate manual backups, which will be always retained unless you delete them.
- If storage space used for manual backups exceeds the default storage space, additional RDS storage costs may incur.

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 **Service List**. Under **Database**, click **Relational Database Service**. The RDS console is displayed.

Step 4 On the **Instance Management** page, click the target DB instance. On the **Backups & Restorations** page, locate the target backup to be replicated and click **Replicate** in the **Operation** column.

Alternatively, choose **Backup Management**. On the displayed page, locate the manual backup to be replicated and choose **More > Replicate** or locate an automated backup and click **Replicate** in the **Operation** column.

Step 5 In the displayed dialog box, enter a new backup name and description and click **OK**.

- The backup name must consist of 4 to 64 characters and start with a letter. It can contain only uppercase letters, lowercase letters, digits, hyphens (-), and underscores (_).
- The description consists of a maximum of 256 characters and cannot contain the following special characters: >!<"&'=

Step 6 After the new backup has been created, you can view and manage it on the **Backup Management** page.

----End

5.3.8 Deleting a Manual Backup


Scenarios

You can delete manual backups to release storage space.

NOTICE

Deleted manual backups cannot be recovered. Exercise caution when performing this operation.

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 **Service List**. Under **Database**, click **Relational Database Service**. The RDS console is displayed.
- Step 4** In the navigation pane on the left, choose **Backup Management**. On the displayed page, locate the target manual backup to be deleted and choose **More > Delete** in the **Operation** column.

The following backups cannot be deleted:

- Automated backups
- Backups that are being restored
- Backups that are being replicated

- Step 5** In the displayed dialog box, click **Yes**.

----End

5.4 Parameter Template Management

5.4.1 Creating a Parameter Template

You can use database parameter templates to manage the DB engine configuration. A database parameter template acts as a container for engine configuration values that can be applied to one or more DB instances.

This default template contains DB engine defaults and system defaults that are configured based on the engine, compute class, and allocated storage of the instance. You cannot modify the parameter settings of a default parameter template. You must create your own parameter template to change parameter settings.

NOTICE

Not all DB engine parameters can be changed in a custom parameter template.

If you want to use your custom parameter template, you simply create a parameter template and select it when you create a DB instance or apply it to an existing DB instance following the instructions provided in section [Applying a Parameter Template](#).

When you have already created a parameter template and want to include most of the custom parameters and values from that template in a new parameter template, you can replicate that parameter template following the instructions provided in section [Replicating a Parameter Template](#).

The following are the key points you should know when using parameters in a parameter template:

- When you change a dynamic parameter value in a parameter template and save the change, the change takes effect immediately. When you change a static parameter value in a parameter template and save the change, the change will take effect only after you manually reboot the DB instances to which the parameter template applies.
- Improper parameter settings may have unintended adverse effects, including degraded performance and system instability. Exercise caution when modifying database parameters and you need to back up data before modifying parameters in a parameter template. Before applying parameter template changes to a production DB instance, you should try out these changes on a test DB instance.


NOTE

RDS does not share parameter template quotas with DDS.

You can create a maximum of 100 parameter templates for RDS DB instances. All RDS DB engines share the parameter template quota.

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 **Service List**. Under **Database**, click **Relational Database Service**. The RDS console is displayed.

Step 4 On the **Parameter Template Management** page, click **Create Parameter Template**.

Step 5 In the displayed dialog box, configure required information and click **OK**.

- Select a DB engine for the parameter template.
- The template name must consist of 1 to 64 characters. It can contain only uppercase letters, lowercase letters, digits, hyphens (-), underscores (_), and periods (.).

- The description consists of a maximum of 256 characters and cannot contain carriage return characters or the following special characters: >!<"&'=

----End

5.4.2 Modifying Parameters

You can modify parameters in a custom parameter template to optimize RDS database performance.

You can change parameter values in custom parameter templates only and cannot change parameter values in default parameter templates.

If you modify a parameter, when the modification takes effect is determined by the type of parameter.

The RDS console displays the statuses of DB instances to which the parameter template applies. For example, if the DB instance has not used the latest modifications made to its parameter template, its status is **Pending reboot**. You need to manually reboot the DB instance for the latest modifications to take effect for that DB instance.

NOTE

RDS has default parameter templates whose parameter values cannot be changed. You can view these parameter values by clicking the default parameter templates. If a custom parameter template is set incorrectly, the database startup may fail. You can re-configure the custom parameter template according to the configurations of the default parameter template.

Modifying Parameter Template Parameters

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 **Service List**. Under **Database**, click **Relational Database Service**. The RDS console is displayed.

Step 4 Choose **Parameter Template Management** in the navigation pane on the left. On the **Custom Templates** page, click the target parameter template.

Step 5 Modify parameters as required.

Relevant parameters are as follows:

- For details on parameter descriptions, visit the [PostgreSQL official website](#).
- If **log_statement** is set to **ddl**, **mod**, or **all**, the operations for creating and deleting database users (including passwords and other sensitive information) are recorded. This operation affects database performance. Exercise caution when setting this parameter.
- The **search_path** parameter must be set to a schema sequence where schemas are separated by commas (,). Ensure that the schemas exist. Otherwise, the database performance will be affected.
- Enabling the following parameters will affect the database performance: **log_hostname**, **log_duration**, **log_connections**, and **log_disconnections**. Exercise caution when enabling these parameters.

- If you enable the parameter **log_duration**, SQL statements containing sensitive information may be recorded in logs. You are advised to disable this parameter.
- If the parameter **log_min_duration_statement** is set to **0**, SQL statements containing sensitive information will be recorded in logs. You are advised to disable this parameter by setting it to **-1**.
- The parameters **nls_timestamp_format** and **nls_date_format** control the input and output formats of timestamp and date. They support PostgreSQL Enhanced Edition only.
 - The **nls_timestamp_format** parameter defines the timestamp format. The **nls_date_format** defines the date format. Their functions are the same and you can change either of them.
- The **temp_file_limit** parameter specifies the maximum amount of disk space (in KB) that a session can use for temporary files. It supports PostgreSQL 11 and 12 only. Changing this parameter value is a high-risk operation. Exercise caution when deciding to perform this operation.
 - If the parameter value exceeds the threshold, the DB instance will become unavailable.
 - If the parameter value is changed to a larger value for temporary use but is not changed to the original value after the use, the disk space will be continuously used to store temporary files. If the disk space is used up, services will be interrupted and the DB instance will become unavailable.


Available operations are as follows:

- To save the modifications, click **Save**.
- To cancel the modifications, click **Cancel**.
- To preview the modifications, click **Preview**.

----End

Modifying Instance Parameters

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 **Service List**. Under **Database**, click **Relational Database Service**. The RDS console is displayed.

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

Step 5 In the navigation pane on the left, choose **Parameters**. On the displayed page, modify parameters as required.

Relevant parameters are as follows:

- For details on parameter descriptions, visit the [PostgreSQL official website](#).
- If **log_statement** is set to **ddl**, **mod**, or **all**, the operations for creating and deleting database users (including passwords and other sensitive information) are recorded. This operation affects database performance. Exercise caution when setting this parameter.

- The **search_path** parameter must be set to a schema sequence where schemas are separated by commas (.). Ensure that the schemas exist. Otherwise, the database performance will be affected.
- Enabling the following parameters will affect the database performance: **log_hostname**, **log_duration**, **log_connections**, and **log_disconnections**. Exercise caution when enabling these parameters.
- If you enable the parameter **log_duration**, SQL statements containing sensitive information may be recorded in logs. You are advised to disable this parameter.
- If the parameter **log_min_duration_statement** is set to **0**, SQL statements containing sensitive information will be recorded in logs. You are advised to disable this parameter by setting it to **-1**.
- The parameters **nl_timestamp_format** and **nl_date_format** control the input and output formats of timestamp and date. They support PostgreSQL Enhanced Edition only.
 - The **nl_timestamp_format** parameter defines the timestamp format. The **nl_date_format** defines the date format. Their functions are the same and you can change either of them.
- The **temp_file_limit** parameter specifies the maximum amount of disk space (in KB) that a session can use for temporary files. It supports PostgreSQL 11 and 12 only. Changing this parameter value is a high-risk operation. Exercise caution when deciding to perform this operation.
 - If the parameter value exceeds the threshold, the DB instance will become unavailable.
 - If the parameter value is changed to a larger value for temporary use but is not changed to the original value after the use, the disk space will be continuously used to store temporary files. If the disk space is used up, services will be interrupted and the DB instance will become unavailable.

Available operations are as follows:

NOTICE

After you modify instance parameters, the modifications immediately take effect for the DB instance.

After you modify parameters in a parameter template, you need to view the status of the DB instance to which the parameter template applies. If the status is **Pending reboot**, you must reboot the DB instance for the modifications to take effect.

- If you have modified parameters of a primary DB instance, you need to reboot the primary DB instance for the modifications to take effect. (For primary/standby DB instances, the parameter modifications also apply to the standby DB instance.)
- If you have modified parameters of a read replica, you need to reboot the read replica for the modifications to take effect.

-
- To save the modifications, click **Save**.
 - To cancel the modifications, click **Cancel**.

- To preview the modifications, click **Preview**.

After parameters are modified, you can view parameter change history by referring to section [Viewing Parameter Change History](#).

----End

5.4.3 Exporting a Parameter Template

Scenarios

- You can export a parameter template of a DB instance for future use. You can apply the exported parameter template to DB instances by referring to section [Applying a Parameter Template](#).
- You can export the parameter template information (parameter names, values, and descriptions) of a DB instance to a CSV file for viewing and analysis.

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 **Service List**. Under **Database**, click **Relational Database Service**. The RDS console is displayed.

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

Step 5 In the navigation pane on the left, choose **Parameters**. On the displayed page, click **Export** above the parameter list.

- Exporting to a custom template

In the displayed dialog box, configure required information and click **OK**.

NOTE

- The template name must consist of 1 to 64 characters. It can contain only uppercase letters, lowercase letters, digits, hyphens (-), underscores (_), and periods (.).
- The description consists of a maximum of 256 characters and cannot contain carriage return characters or the following special characters: >!<'&'=

After the parameter template is exported, a new template is generated in the list on the **Parameter Template Management** page.

- Exporting to a file

The parameter template information (parameter names, values, and descriptions) of a DB instance is exported to a CSV file. In the displayed dialog box, enter the file name and click **OK**.

NOTE

The file name must start with a letter and consist of 4 to 64 characters. It can contain only letters, digits, hyphens (-), and underscores (_).

----End


5.4.4 Comparing Parameter Templates

Scenarios


You can compare DB instance parameters with a parameter template that uses the same DB engine to understand the differences of parameter settings.

You can also compare default parameter templates that use the same DB engine to understand the differences of parameter settings.

Comparing Instance Parameters with a Parameter Template

- 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 **Service List**. Under **Database**, click **Relational Database Service**. The RDS console is displayed.
 - Step 4** On the **Instance Management** page, click the target DB instance.
 - Step 5** In the navigation pane on the left, choose **Parameters**. On the displayed page, click **Compare** above the parameter list.
 - Step 6** In the displayed dialog box, select a parameter template to be compared and click **OK**.
 - If their settings are different, the parameter names and values of both parameter templates are displayed.
 - If their settings are the same, no data is displayed.
- End

Comparing Parameter Templates

- 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 **Service List**. Under **Database**, click **Relational Database Service**. The RDS console is displayed.
 - Step 4** On the **Parameter Template Management** page, click **Custom Templates**. Locate the target parameter template and click **Compare** in the **Operation** column.
 - Step 5** In the displayed dialog box, select a parameter template that uses the same DB engine as the target template and click **OK**.
 - If their settings are different, the parameter names and values of both parameter templates are displayed.
 - If their settings are the same, no data is displayed.
- End

5.4.5 Viewing Parameter Change History


Scenarios

You can view the change history of DB instance parameters or custom parameter templates.

 **NOTE**

An exported or custom parameter template has initially a blank change history.

Viewing Change History of DB Instance Parameters


- 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 **Service List**. Under **Database**, click **Relational Database Service**. The RDS console is displayed.
- Step 4** On the **Instance Management** page, click the target DB instance.
- Step 5** In the navigation pane on the left, choose **Parameters**. On the displayed page, click **Change History**.

You can view the parameter name, original parameter value, new parameter value, modification status, modification time, application status, and application time.

You can apply the parameter template to DB instances as required by referring to section [Applying a Parameter Template](#).

----End

Viewing Change History of a Parameter Template

- 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 **Service List**. Under **Database**, click **Relational Database Service**. The RDS console is displayed.
- Step 4** Choose **Parameter Template Management** in the navigation pane on the left. On the **Custom Templates** page, click the target parameter template.
- Step 5** On the displayed page, choose **Change History** in the navigation pane on the left.

You can view the parameter name, original parameter value, new parameter value, modification status, and modification time.

----End

5.4.6 Replicating a Parameter Template

Scenarios

You can replicate a parameter template you have created. When you have already created a parameter template and want to include most of the custom parameters and values from that template in a new parameter template, you can replicate that parameter template.

After a parameter template is replicated, the new template may be displayed about 5 minutes later.

Default parameter templates cannot be replicated. You can create parameter templates based on the default ones.

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 **Service List**. Under **Database**, click **Relational Database Service**. The RDS console is displayed.

Step 4 On the **Parameter Template Management** page, click **Custom Templates**. Locate the target parameter template and click **Replicate** in the **Operation** column.

Alternatively, click the target DB instance on the **Instance Management** page. On the **Parameters** page, click **Export** to generate a new parameter template for future use.

Step 5 In the displayed dialog box, configure required information and click **OK**.

- The template name must consist of 1 to 64 characters. It can contain only uppercase letters, lowercase letters, digits, hyphens (-), underscores (_), and periods (.).
- The description consists of a maximum of 256 characters and cannot contain carriage return characters or the following special characters: >!<"&'=

After the parameter template is replicated, a new template is generated in the list on the **Parameter Template Management** page.

----End


5.4.7 Resetting a Parameter Template

Scenarios

You can reset all parameters in a custom parameter template to their default settings.

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 **Service List**. Under **Database**, click **Relational Database Service**. The RDS console is displayed.
- Step 4** On the **Parameter Template Management** page, click **Custom Templates**. Locate the target parameter template and choose **More > Reset** in the **Operation** column.
- Step 5** Click **Yes**.

 **NOTE**

After you reset the parameter template, click the DB instance to which the parameter template is applied to view the status of the parameter template. On the displayed **Basic Information** page, if the status of the parameter template is **Pending reboot**, you must reboot the DB instance for the reset to take effect.


----End

5.4.8 Applying a Parameter Template

Scenarios

Modifications to parameters in a custom parameter template take effect only after you apply this parameter template to target DB instances. A parameter template can be applied only to DB instances of the same version.

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 **Service List**. Under **Database**, click **Relational Database Service**. The RDS console is displayed.
- Step 4** On the **Parameter Template Management** page, perform the following operations based on the type of the parameter template to be applied:
- If you intend to apply a default parameter template to DB instances, click **Default Templates**, locate the target parameter template, and click **Apply** in the **Operation** column.
 - If you intend to apply a custom parameter template to DB instances, click **Custom Templates**, locate the target parameter template, and choose **More > Apply** in the **Operation** column.

A parameter template can be applied to one or more DB instances.

- Step 5** In the displayed dialog box, select one or more DB instances to which the parameter template will be applied and click **OK**.

After the parameter template is successfully applied, you can view the application records by referring to section [Viewing Application Records of a Parameter Template](#).


----End

5.4.9 Viewing Application Records of a Parameter Template

Scenarios

You can view the application records of a parameter template.

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 **Service List**. Under **Database**, click **Relational Database Service**. The RDS console is displayed.
- Step 4** After a parameter template applies to a DB instance successfully, choose **Parameter Template Management** in the navigation pane on the left, locate the target parameter template, and click **View Application Record** in the **Operation** column on the **Default Templates** page or choose **More > View Application Record** on the **Custom Templates** page.

You can view the name or ID of the DB instance to which the parameter template applies, as well as the application status, application time, and failure cause.

----End

5.4.10 Modifying a Parameter Template Description





Scenarios

You can modify the description of a parameter template you have created.

NOTE

You cannot modify the description of a default parameter template.

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 **Service List**. Under **Database**, click **Relational Database Service**. The RDS console is displayed.
- Step 4** On the **Parameter Template Management** page, click **Custom Templates**. Locate the target parameter template and click  in the **Description** column.
- Step 5** Enter a new description. You can click  to submit or  to cancel the modification.
 - After you submit the modification, you can view the new description in the **Description** column on the **Parameter Template Management** page.

- The description consists of a maximum of 256 characters and cannot contain the following special characters: >!<"&'=

----End

5.4.11 Deleting a Parameter Template

Scenarios

You can delete a custom parameter template that is no longer in use.

NOTICE

- Deleted parameter templates cannot be recovered. Exercise caution when performing this operation.
- Default parameter templates cannot be deleted.

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 **Service List**. Under **Database**, click **Relational Database Service**. The RDS console is displayed.

Step 4 On the **Parameter Template Management** page, click **Custom Templates**. Locate the target parameter template to be deleted and choose **More > Delete** in the **Operation** column.

Step 5 In the displayed dialog box, click **Yes**.

----End

5.5 Connection Management

5.5.1 Configuring and Changing a Floating IP Address

Scenarios


You can plan and change floating IP addresses after migrating on-premises databases or other cloud databases to RDS.

Configuring a Floating IP Address

You can use an automatically-assigned IP address when creating a DB instance.

Changing a Floating IP Address

After a DB instance is created, you can change its floating IP address.

- 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 **Service List**. Under **Database**, click **Relational Database Service**. The RDS console is displayed.
- Step 4** On the **Instance Management** page, click the target DB instance.
- Step 5** In the **Connection Information** area on the **Basic Information** page, click **Change** in the **Floating IP Address** field.
- Step 6** In the displayed dialog box, enter a new floating IP address and click **OK**.

----End

5.5.2 Binding and Unbinding an EIP

Scenarios

By default, a DB instance is not publicly accessible (not bound with an EIP) after being created. You can bind an EIP to the DB instance for public accessibility and can unbind the EIP from the DB instance as required.


NOTICE

To ensure successful access to the database, the security group associated with the database must allow access over the database port. For example, if the database port is 5432, ensure that the security group allow access over the 5432 port.

Prerequisites

- You can bind an EIP to a primary DB instance or a read replica only.
- If a DB instance has already been bound with an EIP, you must unbind the EIP from the DB instance first before binding a new EIP to it.

Binding an Unbinding 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 **Service List**. Under **Database**, click **Relational Database Service**. The RDS console is displayed.
- Step 4** On the **Instance Management** page, click the target DB instance.
- Step 5** On the **EIPs** page, click **Bind EIP**.
- Step 6** In the displayed dialog box, all unbound EIPs are listed. Select the EIP to be bound and click **OK**. If no available EIPs are displayed, click **View EIP** and obtain an EIP.
- Step 7** On the **EIPs** page of the RDS console, 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.

To unbind the EIP from the DB instance, see [Unbinding an EIP](#).

----End

Unbinding 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 **Service List**. Under **Database**, click **Relational Database Service**. The RDS console is displayed.

Step 4 On the **Instance Management** page, click the DB instance that has been bound with an EIP.

Step 5 On the **EIPs** page, locate the target EIP to be unbound and click **Unbind**. In the displayed dialog box, click **Yes**.

Step 6 On the **EIPs** page, view the unbinding result.

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

To bind an EIP to the DB instance again, see [Binding an Unbinding an EIP](#).

----End


5.5.3 Changing the Database Port

Scenarios


This section describes how to change the database port of a primary DB instance or a read replica. For primary/standby DB instances, changing the database port of the primary DB instance will cause the database port of the standby DB instance to also be changed accordingly.


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 **Service List**. Under **Database**, click **Relational Database Service**. The RDS console is displayed.

Step 4 On the **Instance Management** page, click the target DB instance or click  first and then click the target read replica.

Step 5 In the **Connection Information** area on the **Basic Information** page, click  in the **Database Port** field.

NOTE

The PostgreSQL database port ranges from 2100 to 9500.

- To submit the change, click .
 - In the dialog box, click **Yes**.
 - i. If you change the database port of the primary DB instance, that of the standby DB instance will also be changed and both DB instances will be rebooted.
 - ii. If you change the database port of a read replica, the change will not affect other DB instances. Only the read replica will be rebooted.
 - iii. This process takes 1-5 minutes.
 - In the dialog box, click **No** to cancel the modification.

Step 6 View the result of the change on the **Basic Information** page.

----End

5.5.4 Connecting to a DB Instance Through pgAdmin

You can use the pgAdmin client to connect to an RDS DB instance.

NOTICE

- The pgAdmin client only supports access through EIPs.
- The pgAdmin version must be 4 or later.

Preparations

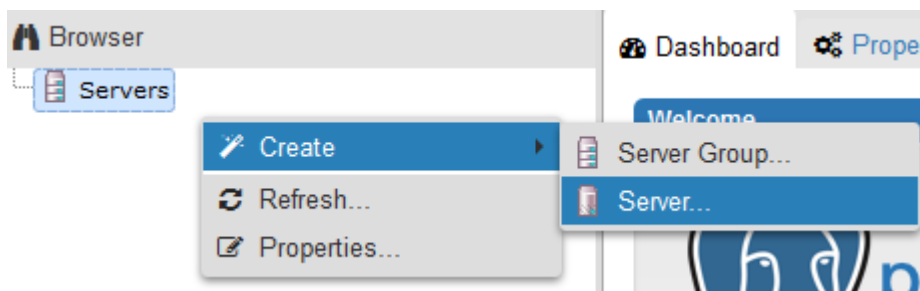
1. Prepare a device that can access RDS DB instances.
To connect to a DB instance through an EIP, you must:
 - a. Bind the EIP to the DB instance. For details, see [Step 2: Bind an EIP](#).
 - b. Ensure that the local device can access the EIP that has been bound to the DB instance.
2. Install the pgAdmin client on the prepared device.

Procedure

Step 1 Start pgAdmin.

Step 2 In the displayed login window, choose **Servers > Create > Server**.

Figure 5-1 Creation



Step 3 On the **General** page, specify **Name**. On the **Connection** page, specify information about the DB instance to be connected. Then, click **Save**.

Figure 5-2 General page

The image shows a 'Create - Server' dialog box with two tabs: 'General' and 'Connection'. The 'General' tab is active. It contains the following fields and controls:

- Name:** A text input field that is currently empty and has a red border, indicating a validation error.
- Server group:** A dropdown menu with 'Servers' selected.
- Connect now?:** A checked checkbox.
- Comments:** A large text area for entering notes.

At the bottom of the dialog, there is a red error message: 'Name must be specified.' Below this message are three buttons: 'Save' (blue), 'Cancel' (red), and 'Reset' (yellow). There are also information and help icons on the left side of the bottom bar.

Figure 5-3 Connection page

The screenshot shows a 'Create - Server' dialog box with a 'Connection' tab. The fields are: Host name/address (empty), Port (5432), Maintenance database (postgres), User name (empty), Password (empty), Save password? (checkbox), Role (empty), and SSL mode (Prefer). A red error bar at the bottom says 'Name must be specified.' Buttons for Save, Cancel, and Reset are at the bottom right.

Parameter description:

- **Host name/address:** indicates the EIP of the DB instance to be connected.
- **Port:** indicates the database port. By default, the value is **5432**.
- **User name:** indicates the username. By default, the value is **root**.
- **Password:** indicates the password of the target database username.

Step 4 In the login window, check that the connection information is correct. The target DB instance is successfully connected.

----End

5.6 Data Migration

5.6.1 Migrating Data to RDS for PostgreSQL Using psql

Preparing for Data Migration

PostgreSQL supports logical backups. You can use the `pg_dump` logical backup function to export backup files and then import them to RDS using `psql`.

You can access RDS DB instances through an EIP or through an ECS.

Preparations

1. Prepare an ECS for accessing DB instances in the same VPC or prepare a device for accessing RDS through an EIP.
 - To connect to a DB instance through an ECS, you must first create an ECS.
 - To connect to a DB instance through an EIP, you must:
 - i. Bind an EIP to a DB instance. For details, see [Binding an EIP](#).
 - ii. Ensure that the local device can access the EIP that has been bound to the DB instance.
2. Install the PostgreSQL client on the prepared ECS or device.

NOTE

The PostgreSQL client version must be the same as the version of RDS for PostgreSQL. The PostgreSQL database or client will provide `pg_dump` and `psql`.

Exporting Data

Before migrating an existing PostgreSQL database to RDS, you need to export the PostgreSQL database.

NOTICE

- The export tool must match the DB engine version.
 - Database migration is performed offline. Before the migration, you must stop any applications using the source database.
-

Step 1 Log in to the ECS or the device that can access RDS.

Step 2 Use the `pg_dump` tool to export the source database into an SQL file.

```
pg_dump --username=<DB_USER> --host=<DB_ADDRESS> --port=<DB_PORT> --  
format=plain --file=<BACKUP_FILE> <DB_NAME>
```

- ***DB_USER*** indicates the database username.
- ***DB_ADDRESS*** indicates the database address.
- ***DB_PORT*** indicates the database port.
- ***BACKUP_FILE*** indicates the name of the file to which the data will be exported.
- ***DB_NAME*** indicates the name of the database to be migrated.

Enter the database password as prompted.

Example:

```
$ pg_dump --username=root --host=192.168.151.18 --port=5432 --  
format=plain --file=backup.sql my_db
```

Password for user root:

After this command is executed, a **backup.sql** file will be generated as follows:

```
[rds@localhost ~]$ ll backup.sql
```



```
-rw-r-----. 1 rds rds 2714 Sep 21 08:23 backup.sql
```

----End

Importing Data

You can connect your client to RDS and import exported SQL files into RDS.

Step 1 Ensure that the destination database to which data is to be imported exists.

If the destination database does not exist, run the following command to create a database:

```
# psql --host=<RDS_ADDRESS> --port=<DB_PORT> --username=root --dbname=postgres -c "create database <DB_NAME>;"
```

- **RDS_ADDRESS** indicates the IP address of the RDS DB instance.
- **DB_PORT** indicates the RDS DB instance port.
- **DB_NAME** indicates the name of the database to be imported.

Step 2 Import the exported file to RDS.

```
# psql --host=<RDS_ADDRESS> --port=<DB_PORT> --username=root --dbname=<DB_NAME> --file=<BACKUP_DIR>/backup.sql
```

- **RDS_ADDRESS** indicates the IP address of the RDS DB instance.
- **DB_PORT** indicates the RDS DB instance port.
- **DB_NAME** indicates the name of the database to which data is to be imported. Ensure that the database exists.
- **BACKUP_DIR** indicates the directory where the **backup.sql** file is stored.

Enter the password for the RDS DB instance as prompted.

Example:

```
# psql --host=172.16.66.198 --port=5432 --username=root --dbname=my_db --file=backup.sql
```

Password for user root:

Step 3 View the import result.

```
my_db=> \l my_db
```

In this example, the database named **my_db** has been imported.

```
my_db=> \l my_db
List of databases
Name | Owner | Encoding | Collate | Ctype | Access privileges
-----+-----+-----+-----+-----+-----
my_db | root | UTF8 | en_US.UTF-8 | en_US.UTF-8 |
(1 row)
```

----End

5.7 Data Security

5.7.1 Resetting the Administrator Password

Scenarios

You can reset the administrator password of a primary instance.

You can also reset the password of your database account when using RDS.

You cannot reset the administrator password under the following circumstances:

- The database port is being changed.
- The status of the primary DB instance is **Creating**, **Restoring**, **Rebooting**, **Storage full**, **Changing port**, or **Abnormal**.

NOTE

- If you have changed the administrator password of the primary DB instance, the administrator passwords of the standby DB instance and read replica (if any) will also be changed accordingly.
- The length of time it takes for the new password to take effect depends on the amount of service data currently being processed by the primary DB instance.
- To prevent brute force cracking and ensure system security, change your password periodically, such as every three or six months.

Method 1

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 **Service List**. Under **Database**, click **Relational Database Service**. The RDS console is displayed.

Step 4 On the **Instance Management** page, locate the target DB instance and choose **More > Reset Password** in the **Operation** column.

Step 5 Enter a new password and confirm the password.

NOTICE


Keep this password secure. The system cannot retrieve it.

The new password must consist of 8 to 32 characters and contain at least three of the following: uppercase letters, lowercase letters, digits, and special characters (~!@#%^*_+?.,). Enter a strong password and periodically change it to improve security, preventing security risks such as brute force cracking.

- To submit the new password, click **OK**.
- To cancel the reset operation, click **Cancel**.

----End

Method 2

- 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 **Service List**. Under **Database**, click **Relational Database Service**. The RDS console is displayed.
- Step 4** On the **Instance Management** page, click the target DB instance.
- Step 5** In the **DB Information** area on the **Basic Information** page, click **Reset Password** in the **Administrator** field.
- Step 6** Enter a new password and confirm the password.

NOTICE

Keep this password secure. The system cannot retrieve it.

The new password must consist of 8 to 32 characters and contain at least three of the following: uppercase letters, lowercase letters, digits, and special characters (~!@#%^*_+?,.). Enter a strong password and periodically change it to improve security, preventing security risks such as brute force cracking.

- To submit the new password, click **OK**.
- To cancel the reset operation, click **Cancel**.




----End


5.7.2 Changing a Security Group


Scenarios

This section describes how to change the security group of a primary DB instance or read replica. For primary/standby DB instances, changing the security group of the primary DB instance will cause the security group of the standby DB instance to also be changed accordingly.

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 **Service List**. Under **Database**, click **Relational Database Service**. The RDS console is displayed.
- Step 4** On the **Instance Management** page, click the target DB instance or read replica.
- Step 5** In the **Connection Information** area on the **Basic Information** page, click  in the **Security Group** field.
 - To submit the change, click .

- To cancel the change, click .

Step 6 Changing the security group takes 1 to 3 minutes. Click  in the upper right corner on the **Basic Information** page to view the result of the change.

----End

5.8 Metrics and Alarms

5.8.1 Configuring Displayed Metrics

The Agent of an RDS DB instance monitors the metrics and status of the DB instance only and does not collect other data except the monitoring metrics.

Description

This section describes the metrics that can be monitored by Cloud Eye as well as their namespaces and dimensions. You can use APIs provided by Cloud Eye to query the monitoring metrics and alarms generated for RDS.

Namespace

SYS.RDS

DB Instance Monitoring Metrics

- [Table 5-4](#) lists the performance metrics of PostgreSQL databases.

Table 5-4 Database performance metrics

Metric	Name	Description	Value Range	Monitored Object	Monitoring Interval (Raw Data)
rds001_cpu_util	CPU Usage	CPU usage of the monitored object	0-100%	Monitored object: ECS Monitored instance type: PostgreSQL instance	1 minute
rds002_mem_util	Memory Usage	Memory usage of the monitored object	0-100%	Monitored object: ECS Monitored instance type: PostgreSQL instance	1 minute

Metric	Name	Description	Value Range	Monitored Object	Monitoring Interval (Raw Data)
rds003_iops	IOPS	Average number of I/O requests processed by the system in a specified period	≥ 0 counts/s	Monitored object: ECS Monitored instance type: PostgreSQL instance	1 minute
rds004_bytes_in	Network Input Throughput	Incoming traffic in bytes per second	≥ 0 bytes/s	Monitored object: ECS Monitored instance type: PostgreSQL instance	1 minute
rds005_bytes_out	Network Output Throughput	Outgoing traffic in bytes per second	≥ 0 bytes/s	Monitored object: ECS Monitored instance type: PostgreSQL instance	1 minute
rds039_disk_util	Storage Space Usage	Storage space usage of the monitored object	0-100%	Monitored object: ECS Monitored instance type: PostgreSQL instance	1 minute
rds040_transaction_logs_usage	Transaction Logs Usage	Storage space usage of transaction logs	≥ 0 MB	Monitored object: database Monitored instance type: PostgreSQL instance	1 minute
rds041_replication_slot_usage	Replication Slot Usage	Storage space usage of replication slot files	≥ 0 MB	Monitored object: database Monitored instance type: PostgreSQL instance	1 minute

Metric	Name	Description	Value Range	Monitored Object	Monitoring Interval (Raw Data)
rds042_database_connections	Database Connections in Use	Number of database connections in use	≥ 0 counts	Monitored object: database Monitored instance type: PostgreSQL instance	1 minute
rds043_maximum_used_transaction_ids	Maximum Used Transaction IDs	Maximum number of transaction IDs that have been used	≥ 0 counts	Monitored object: database Monitored instance type: PostgreSQL instance	1 minute
rds044_transaction_logs_generations	Transaction Logs Generation	Size of transaction logs generated per second	≥ 0 MB/s	Monitored object: database Monitored instance type: PostgreSQL instance	1 minute
rds045_oldest_replication_slot_lag	Oldest Replication Slot Lag	Lagging size of the most lagging replica in terms of WAL data received	≥ 0 MB	Monitored object: database Monitored instance type: PostgreSQL instance	1 minute
rds046_replication_lag	Replication Lag	Replication lag	≥ 0 ms	Monitored object: database Monitored instance type: PostgreSQL instance	1 minute
rds047_disk_total_size	Total Storage Space	Total storage space of the monitored object	40–4,000 GB	Monitored object: database Monitored instance type: PostgreSQL instance	1 minute

Metric	Name	Description	Value Range	Monitored Object	Monitoring Interval (Raw Data)
rds048_disk_used_size	Used Storage Space	Used storage space of the monitored object	0–4,000 GB	Monitored object: database Monitored instance type: PostgreSQL instance	1 minute
rds049_disk_read_throughput	Disk Read Throughput	Number of bytes read from the disk per second	≥ 0 bytes/s	Monitored object: database Monitored instance type: PostgreSQL instance	1 minute
rds050_disk_write_throughput	Disk Write Throughput	Number of bytes written into the disk per second	≥ 0 bytes/s	Monitored object: database Monitored instance type: PostgreSQL instance	1 minute
rds051_avg_disk_sec_per_read	Average Time per Disk Read (to be deprecated)	Average time required for each disk read in a specified period	> 0s	Monitored object: database Monitored instance type: PostgreSQL instance	1 minute
rds052_avg_disk_sec_per_write	Average Time per Disk Write (to be deprecated)	Average time required for each disk write in a specified period	> 0s	Monitored object: database Monitored instance type: PostgreSQL instance	1 minute

Dimension

Key	Value
postgresql_instance_id	PostgreSQL DB instance ID

5.8.2 Setting Alarm Rules

Scenarios

You can set alarm rules by referring to [Setting Alarm Rules](#) to customize the monitored objects and notification policies and stay aware of the RDS operating status.

The RDS alarm rules include alarm rule names, services, dimensions, monitored objects, metrics, alarm thresholds, monitoring period, and whether to send notifications.

Setting Alarm Rules

- Step 1** Log in to the management console.
- Step 2** Click **Service List** and choose **Management & Deployment > Cloud Eye**.
- Step 3** In the navigation pane on the left, choose **Alarm Management > Alarm Rules**.
- Step 4** On the displayed **Alarm Rules** page, click **Create Alarm Rule**.

----End

5.8.3 Viewing Monitoring Metrics

Scenarios

The Cloud Eye service monitors operating statuses of RDS DB instances. You can view the RDS monitoring metrics on the management console.

Monitored data takes some time for transmission and display. The RDS status displayed on the Cloud Eye console is the status of the last 5 to 10 minutes. If your RDS DB instance is newly created, wait for 5 to 10 minutes and then view the monitoring data.

Prerequisites

- RDS is running properly.
Monitoring metrics of the RDS DB instances that are faulty or have been deleted cannot be displayed on the Cloud Eye console. You can view their monitoring metrics after they are rebooted or restored to be normal.


NOTE

If an RDS DB instance has been faulty for 24 hours, Cloud Eye considers that it does not exist and deletes it from the monitoring object list. You need to manually clear the alarm rules created for the DB instance.

- RDS keeps running properly for about 10 minutes.
For a newly created RDS DB instance, you need to wait for a while before viewing the monitoring metrics.

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 **Service List**. Under **Database**, click **Relational Database Service**. The RDS console is displayed.

Step 4 On the **Instance Management** page, locate the target DB instance and click **View Metric** in the **Operation** column to go to the Cloud Eye console.

Alternatively, click the target DB instance. On the displayed page, click **View Metric** in the upper right corner of the page to go to the Cloud Eye console.

Step 5 On the Cloud Eye console, view monitoring metrics of the primary DB instance.

Cloud Eye can monitor performance metrics from the last 1 hour, 3 hours, 12 hours, 1 day, 7 days, and 30 days.

----End

5.9 Log Management

5.9.1 Viewing Slow Query Logs

Scenarios

Slow query logs record statements that exceed the **log_min_duration_statement** value. You can view log details to identify statements that are slowly executed and optimize the statements.

RDS supports the following statement types:


- SELECT
- INSERT
- UPDATE
- DELETE
- CREATE
- DROP
- ALTER
- DO
- CALL
- COPY

Parameter Description

Table 5-5 Parameters related to PostgreSQL slow queries

Parameter	Description
log_min_duration_statement	Specifies the minimum execution time. The statements whose execution time is greater than or equal to the value of this parameter are recorded.

Viewing Log Details

- 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 **Service List**. Under **Database**, click **Relational Database Service**. The RDS console is displayed.
- Step 4** On the **Instance Management** page, click the target DB instance.
- Step 5** In the navigation pane on the left, choose **Logs**. On the **Slow Query Logs** page, view details about slow query logs.
 - You can view the slow query log records of a specified execution statement type or a specific time period.
 - The **log_min_duration_statement** parameter determines when a slow query log is recorded. However, changes to this parameter do not affect already recorded logs. If **log_min_duration_statement** is changed from 1s to 0.1s, none of the previously recorded logs that do not meet the new threshold are deleted. For example, a 1.5s SQL statement that was recorded when the threshold was 1s will not be deleted now that the new threshold is 2s.

----End

5.10 Task Center

5.10.1 Viewing a Task

You can view the detailed progress and result of the task on the **Task Center** page.


 **NOTE**

You can view and manage the following tasks:


- Creating DB instances
- Rebooting DB instances
- Binding EIPs to DB instances
- Unbinding EIPs from DB instances
- Switching primary/standby DB instances
- Changing single DB instances to primary/standby
- Scaling up storage space
- Creating read replicas
- Restoring data to new DB instances


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 **Service List**. Under **Database**, click **Relational Database Service**. The RDS console is displayed.

Step 4 On the **Task Center** page, locate the target task and click  in front of it to show its details.

- To identify the target task, you can use the task name and DB instance name/ID or enter the target task name in the search box in the upper right corner.
- You can click  in the upper right corner to view the progress and status of tasks in a specific period. The default period is seven days.
A task can be retained for a maximum of one month.
- You can view the instant tasks in the following statuses:
 - Running
 - Completed
 - Failed

----End


5.10.2 Deleting a Task Record

You can delete the task records no longer need to be displayed. The deletion only deletes the task records, and does not delete the DB instances or terminate the tasks that are being executed.

NOTICE

Deleted task records cannot be recovered. Exercise caution when performing this operation.

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 **Service List**. Under **Database**, click **Relational Database Service**. The RDS console is displayed.
- Step 4** Choose **Task Center** in the navigation pane on the left. On the displayed page, locate the target task record to be deleted and click **Delete** in the **Operation** column. In the displayed dialog box, click **Yes**.

You can delete the records of instant tasks in any of the following statuses:

- Running
- Completed
- Failed

----End

5.11 Plugin Management

5.11.1 Creating and Deleting a Plugin

RDS provides the PostgreSQL plugin management solution for user **root**. The `auto_explain` plugin is automatically created by the system and other plugins need to be manually created.

NOTE

The PostgreSQL plugin takes effect at the database level, not globally. You need to manually create it on corresponding databases.

Creating a Plugin

- Step 1** Connect to the database **database1** as user **root** and use **template1** to create a database that can support the plugin.

```
# psql --host=<RDS_ADDRESS> --port=<DB_PORT> --dbname=database1 --
username=root -c "create database <DB_NAME> template template1;"
```

- **RDS_ADDRESS** indicates the IP address of the RDS DB instance.
- **DB_PORT** indicates the RDS DB instance port.
- **DB_NAME** indicates the name of the database to be created.

Enter the password of user **root** as prompted.

Create a database named **my_extension_db** that can support the plugin. Example:

```
# psql --host=192.168.6.141 --port=5432 --dbname=database1 --
username=root -c "create database my_extension_db template template1;"
```

```
Password for user root:
CREATE DATABASE
```

Note: If you are creating a database as a common user, log in to the created database as the common user and run the following command to grant all rights to user **root**:

GRANT ALL ON DATABASE db1 TO root;

Step 2 Connect to the created database as user **root** and create a plugin.

```
# psql --host=<RDS_ADDRESS> --port=<DB_PORT> --dbname=<DB_NAME> --
username=root -c "select control_extension('create', <EXTENSION_NAME>);"
```

- **RDS_ADDRESS** indicates the IP address of the RDS DB instance.
- **DB_PORT** indicates the RDS DB instance port.
- **DB_NAME** indicates the name of the database to be created.
- **EXTENSION_NAME** indicates the plugin name. For more information, see [Plugins Supported By RDS for PostgreSQL](#).

Enter the password of user **root** as prompted.

Create the postgis plugin in the database **my_extension_db**. Example:

```
# psql --host=192.168.6.141 --port=5432 --dbname=my_extension_db --
username=root -c "select control_extension('create','postgis');"
```

```
Password for user root:
control_extension
-----
create postgis successfully.
(1 row)
```

----End

Deleting a Plugin

Connect to the database with a plugin created as user **root** and delete the plugin.

```
# psql --host=<RDS_ADDRESS> --port=<DB_PORT> --username=root --
dbname=<DB_NAME> -c "select control_extension
('drop', <EXTENSION_NAME>);"
```

- **RDS_ADDRESS** indicates the IP address of the RDS DB instance.
- **DB_PORT** indicates the RDS DB instance port.
- **DB_NAME** indicates the name of the database to be created.
- **EXTENSION_NAME** indicates the plugin name. For more information, see [Plugins Supported By RDS for PostgreSQL](#).

Enter the password of user **root** as prompted.

Example:

```
# psql --host=192.168.6.141 --port=5432 --dbname=my_extension_db --
username=root -c "select control_extension('drop','postgis');"
```

```
Password for user root:
control_extension
-----
drop postgis successfully.
(1 row)
```

5.11.2 Plugins Supported By RDS for PostgreSQL

 NOTE

The following table lists the plugins supported by the latest minor versions of RDS for PostgreSQL. If the current PostgreSQL version does not support a plug-in, contact customer service to upgrade the PostgreSQL minor version to the latest.

Table 5-6 Supported plugins

Plugin Name	PostgreSQL 9.5	PostgreSQL 9.6	PostgreSQL 10	PostgreSQL 11	PostgreSQL Enhanced Edition	PostgreSQL 12
address_standardizer	2.5.1	2.5.1	2.5.1	2.5.1	2.5.1	3.0.0
address_standardizer_data_us	2.5.1	2.5.1	2.5.1	2.5.1	2.5.1	3.0.0
btree_gin	1.0	1.0	1.2	1.3	1.3	1.3
btree_gist	1.1	1.2	1.5	1.5	1.5	1.5
cube	1.0	1.2	1.2	1.4	1.4	1.4
dict_int	1.0	1.0	1.0	1.0	1.0	1.0
dict_xsyn	1.0	1.0	1.0	1.0	1.0	1.0
earthdistance	1.0	1.1	1.1	1.1	1.1	1.1
fuzzystrmatch	1.0	1.1	1.1	1.1	1.1	1.1
hll	2.12	2.12	2.12	2.12	2.12	2.14
hstore	1.3	1.4	1.4	1.5	1.5	1.6
intagg	1.0	1.1	1.1	1.1	1.1	1.1
intarray	1.0	1.2	1.2	1.2	1.2	1.2
ltree	1.0	1.1	1.1	1.1	1.1	1.1
ora_migrator	-	-	0.9.3	0.9.3	0.9.3	-
pg_cron	-	-	-	-	-	1.2
pg_trgm	1.1	1.3	1.3	1.4	1.4	1.4
pgcrypto	1.2	1.3	1.3	1.3	1.3	1.3
pgstattuple	-	-	1.5	1.5	1.5	1.5
plpgsql	1.0	1.0	1.0	1.0	1.0	1.0
postgis	2.5.1	2.5.1	2.5.1	2.5.1	2.5.1	3.0.0

Plugin Name	PostgreSQL 9.5	PostgreSQL 9.6	PostgreSQL 10	PostgreSQL 11	PostgreSQL Enhanced Edition	PostgreSQL 12
postgres_fdw	-	-	-	1.0	1.0	1.0
postgis_raster	Integrated to postgis	Integrated to postgis	Integrated to postgis	Integrated to postgis	Integrated to postgis	-
postgis_tiger_geocoder	2.5.1	2.5.1	2.5.1	2.5.1	2.5.1	3.0.0
postgis_topology	2.5.1	2.5.1	2.5.1	2.5.1	2.5.1	3.0.0
tablefunc	1.0	1.0	1.0	1.0	1.0	1.0
timescaledb	-	1.3.2	1.3.2	1.3.2	1.3.2	1.7.0
unaccent	1.0	1.1	1.1	1.1	1.1	1.1
uuid-oss	1.0	1.1	1.1	1.1	1.1	1.1
zhparser	1.0	1.0	1.0	1.0	1.0	1.0

Plugin Description

- **postgis**
 - When postgis is created, the following plugins are created at the same time:
 - postgis
 - postgis_topology
 - fuzzystrmatch
 - postgis_tiger_geocoder
 - address_standardizer
 - address_standardizer_data_us
 - After the postgis plugin is created on the primary DB instance, you need to disconnect the connection from the standby DB instance first and re-establish a connection to update the **search_path** setting.
 - For PostgreSQL Enhanced Edition DB instances, you need to set **empty_is_null** to **OFF** on the console before creating the postgis plugin.
- **earthdistance**

To install the earthdistance plugin, you must install the cube plugin first.
- **cube**

If the earthdistance plugin has been installed, deleting the cube plugin will cause the earthdistance plugin to be unavailable.

- **timescaledb**
The timescaledb plugin does not support the TLS protocol. For more information, see [APIs Not Supported by the timescaledb Plugin](#).
- pgstattuple
Only PostgreSQL 10, 11, and 12 are supported.

APIs Not Supported by the timescaledb Plugin

- add_compress_chunks_policy
- add_drop_chunks_policy
- add_reorder_policy
- alter_job_schedule
- compress_chunk
- decompress_chunk
- drop_chunks
- interpolate
- locf
- move_chunk
- remove_compress_chunks_policy
- remove_drop_chunks_policy
- remove_reorder_policy
- reorder_chunk
- set_integer_now_func
- time_bucket_gapfill

6 FAQs

6.1 Product Consulting

6.1.1 What Precautions Should Be Taken When Using RDS?

1. DB instances' operating systems (OSs) are invisible to you. Your applications can access a database only through the IP address and port.
2. The backup files stored in OBS and the ECS used by RDS are invisible to you. They are visible only in the RDS instance management system.
3. Precautions after purchasing RDS:

After purchasing RDS DB instances, you do not need to perform basic database O&M operations, such as applying HA and security patches. However, you must still pay attention to:

- a. Whether the CPU, input/output operations per second (IOPS), and space are insufficient for the RDS DB instances. If any of these becomes insufficient, you will need to change the CPU/memory or scale up the DB instance.
- b. Whether the performance of the RDS DB instances is adequate, a large number of slow query SQL statements exist, SQL statements need to be optimized, or any indexes are redundant or missing.

6.1.2 What Is the Availability of RDS DB Instances?

Formula for an RDS DB instance availability:

DB instance availability = $(1 - \text{Failure duration} / \text{Total service duration}) \times 100\%$

6.1.3 Can I Use a Template to Create DB Instances?

Currently, you cannot use a template to create DB instances.

6.1.4 What Are the Differences Between RDS and Other Database Solutions?

Table 6-1 Differences between RDS and other database solutions

Function Item	RDS	Self-Built Database Service
Service availability	For details, see Elastic Cloud Service User Guide.	Requires self-guarantee, primary/standby relationship setup, and RAID setup.
Data reliability	For more information, see the <i>Elastic Volume Service User Guide</i> .	Requires self-guarantee, primary/standby relationship setup, and RAID setup.
System security	Defends against Anti-DDoS attacks and promptly repairs database security vulnerabilities.	Requires procurement of expensive devices and software, as well as manual detection and repair of security vulnerabilities.
Database backup	Automated backups	You must find backup storage space to back up the database by yourself and periodically check whether backup data can be restored.
Hardware and software investment	Supports on-demand pricing and scaling without requiring hardware and software investment.	Requires large investment in database servers.
System hosting	Not required.	The hosting cost is high.
Maintenance cost	Not required.	Full-time Database Administrators (DBAs) are required for maintenance, leading to high labor costs.
Deployment and scaling	Supports elastic scaling, fast deployment, and on-demand enabling.	Requires procurement, deployment, and coordination of hardware.
Resource utilization	Bills users based on the resources actually used, resulting in high resource utilization.	Peak resource utilization is considered, leading to low resource usage.

6.1.5 Will My RDS DB Instances Be Affected by Other Users' DB Instances?

No. Your RDS DB instances and resources are isolated from other users' DB instances.

6.1.6 Does RDS Support Cross-AZ High Availability?

Yes. RDS supports cross-AZ high availability. When you purchase primary/standby DB instances, you can select different AZs for them.

6.1.7 Can RDS Primary/Standby DB Instances Be Changed to Single DB Instances?

No. Only RDS single DB instances can be changed to primary/standby DB instances.

6.1.8 What Should I Do If Garbled Characters Are Displayed After SQL Query Results Are Exported to an Excel File?

The default code is utf8. You need to convert the default code to Unicode.

6.1.9 How Do I Create an AD Domain?

Active Directory, which is short for AD, is a directory service on Windows Standard Server, Windows Enterprise Server, and Windows Datacenter Server. (Active Directory cannot run on the Windows Web Server, but it can manage the computers running the Windows Web Server.) Active Directory stores information about objects on the network and makes this information easy for administrators and users to find and use. Active Directory uses a structured data store as the basis for a logical, hierarchical organization of directory information.

Procedure

This section describes how to use Windows Server 2012 R2 to create a domain server.

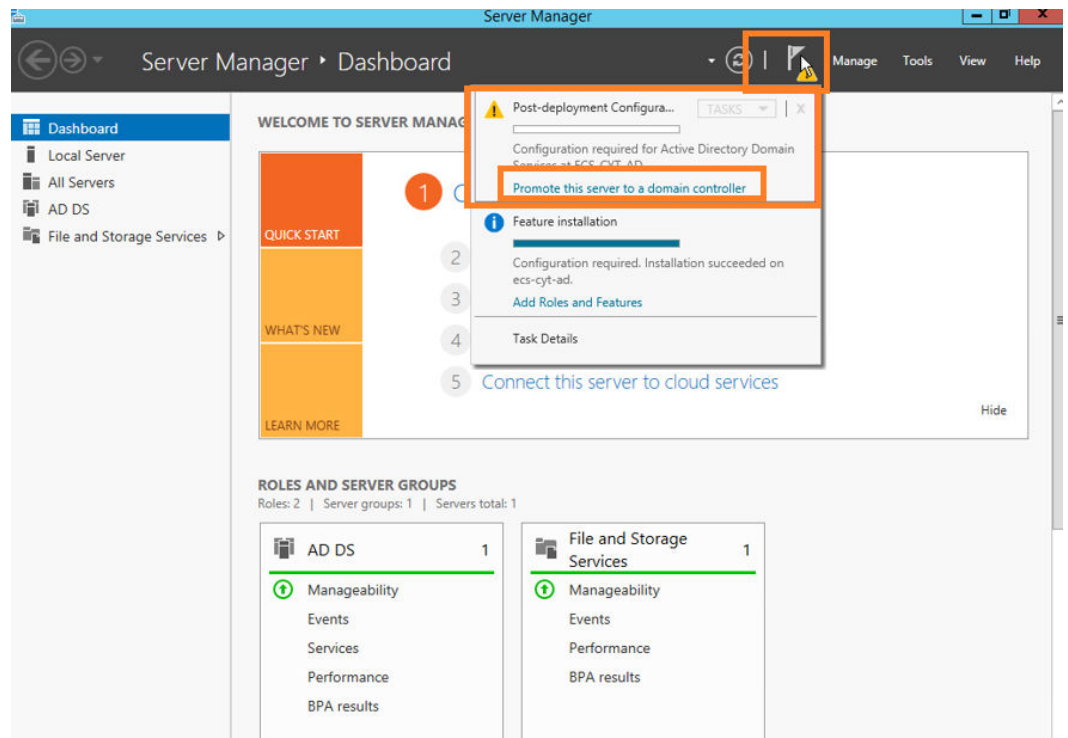
NOTE

1. When you configure an AD domain information during the DB instance creation, do not configure or disable Group Policy Object (GPO) for your domain controller server. Otherwise, the DB instance creation will fail.
2. If GPO is required, you need to create an ECS and set up a new domain controller server with GPO disabled. Then, establish trust between your domain controller server and the new domain controller server. For details, contact customer service.

Step 1 Install an AD domain controller.

1. In Server Manager, choose **Manage > Add Roles and Features**.

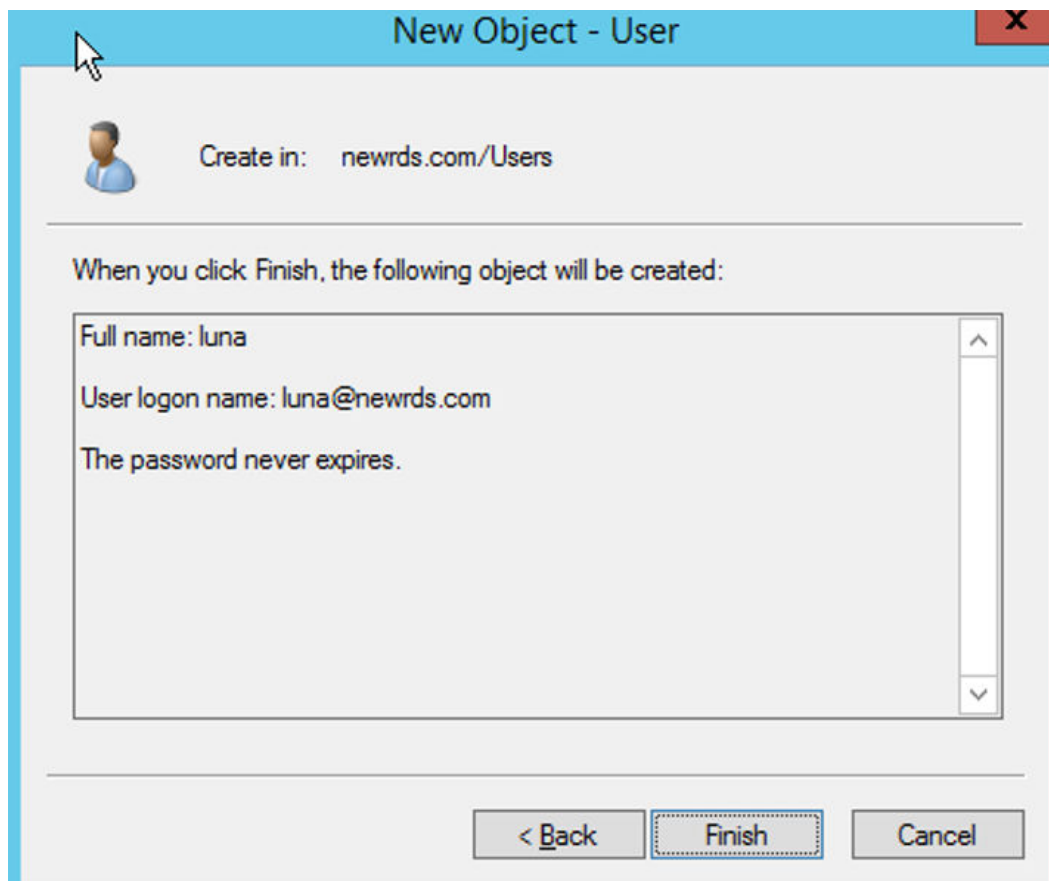
2. In the **Add Roles and Features Wizard** dialog box, click **Next** until the **Select server roles** page is displayed. Select **Active Directory Domain Services** and click **Add Features** in the displayed box.
3. Click **Next** until the **Confirm installation selections** page is displayed. Click **Install** to start the role installation process.
4. After the installation is complete, a yellow triangle icon is displayed. Click **Promote this server to a domain controller**. The **Active Directory Domain Services Configuration Wizard** window is displayed.



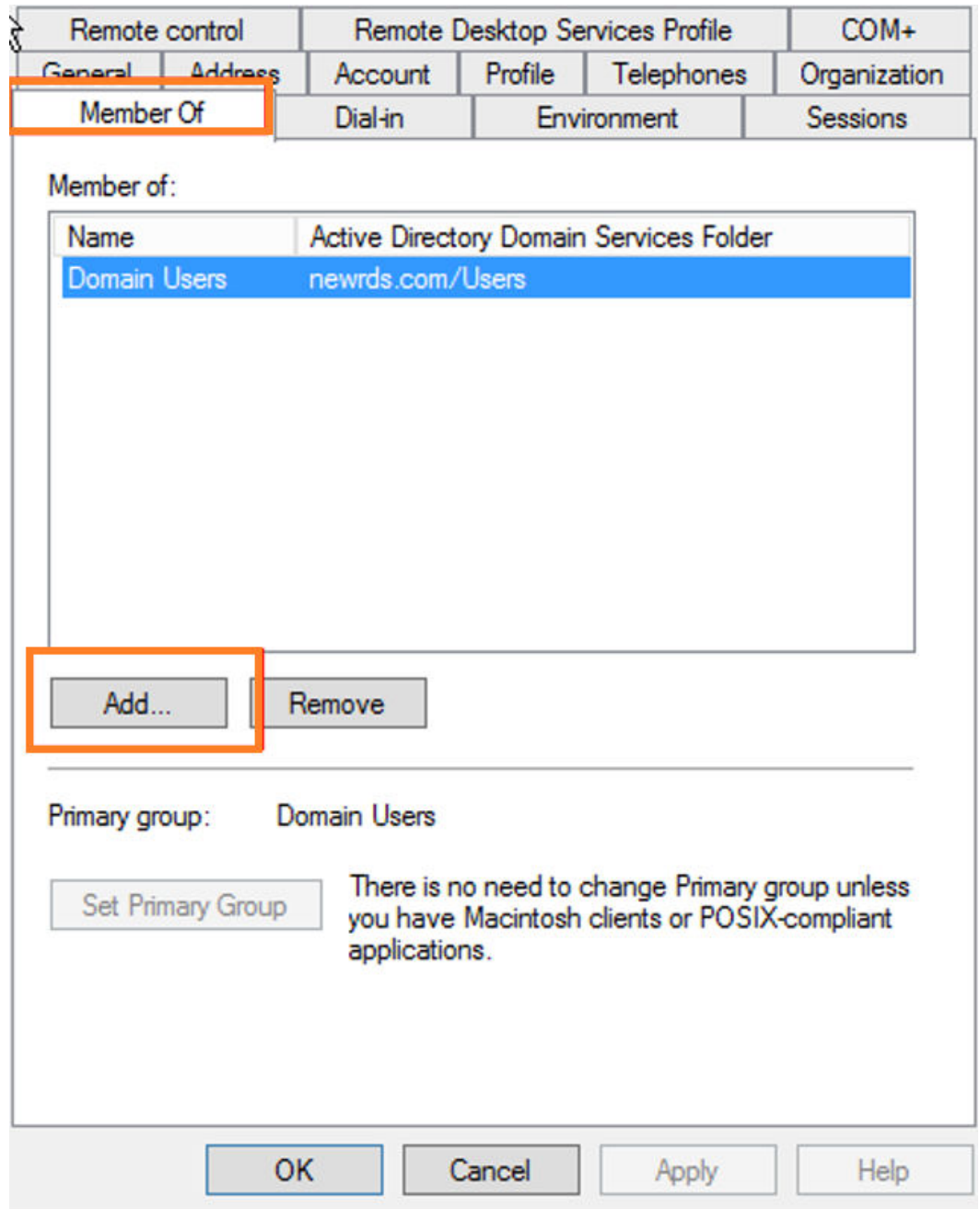
5. On the **Deployment Configuration** page, select **Add a new forest** and set a domain name, such as newrds.com.
6. Click **Next**. On the displayed page, enter the DSRM password (non-domain user).
7. Click **Next** until the **Prerequisites Check** page displayed. Click **Install**. After the installation is complete, the server automatically reboots.
8. Modify the DNS configuration of the network interface. Set the IP address of the active DNS server to the server's private IP address, such as 192.168.0.133.

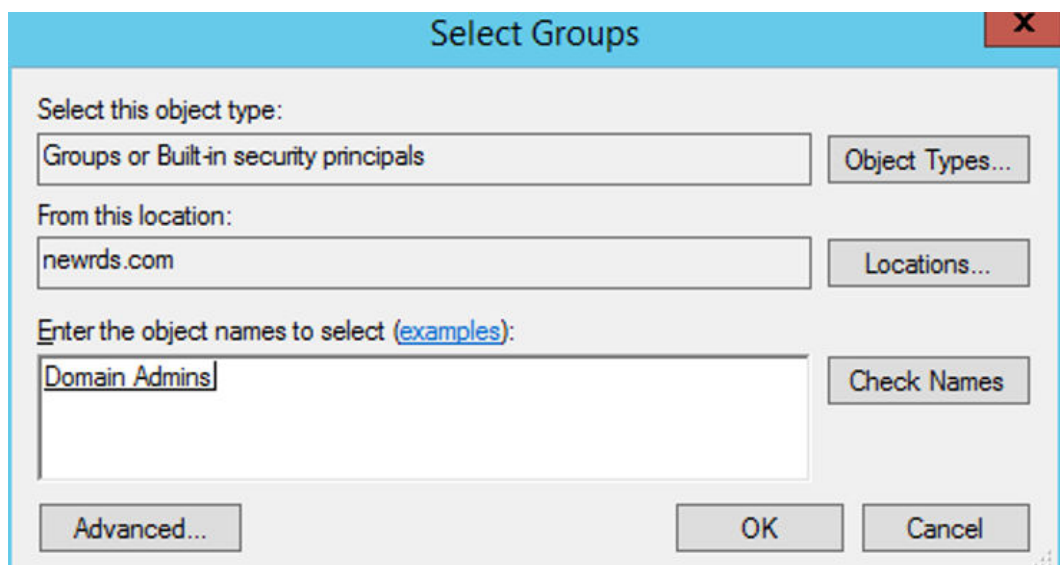
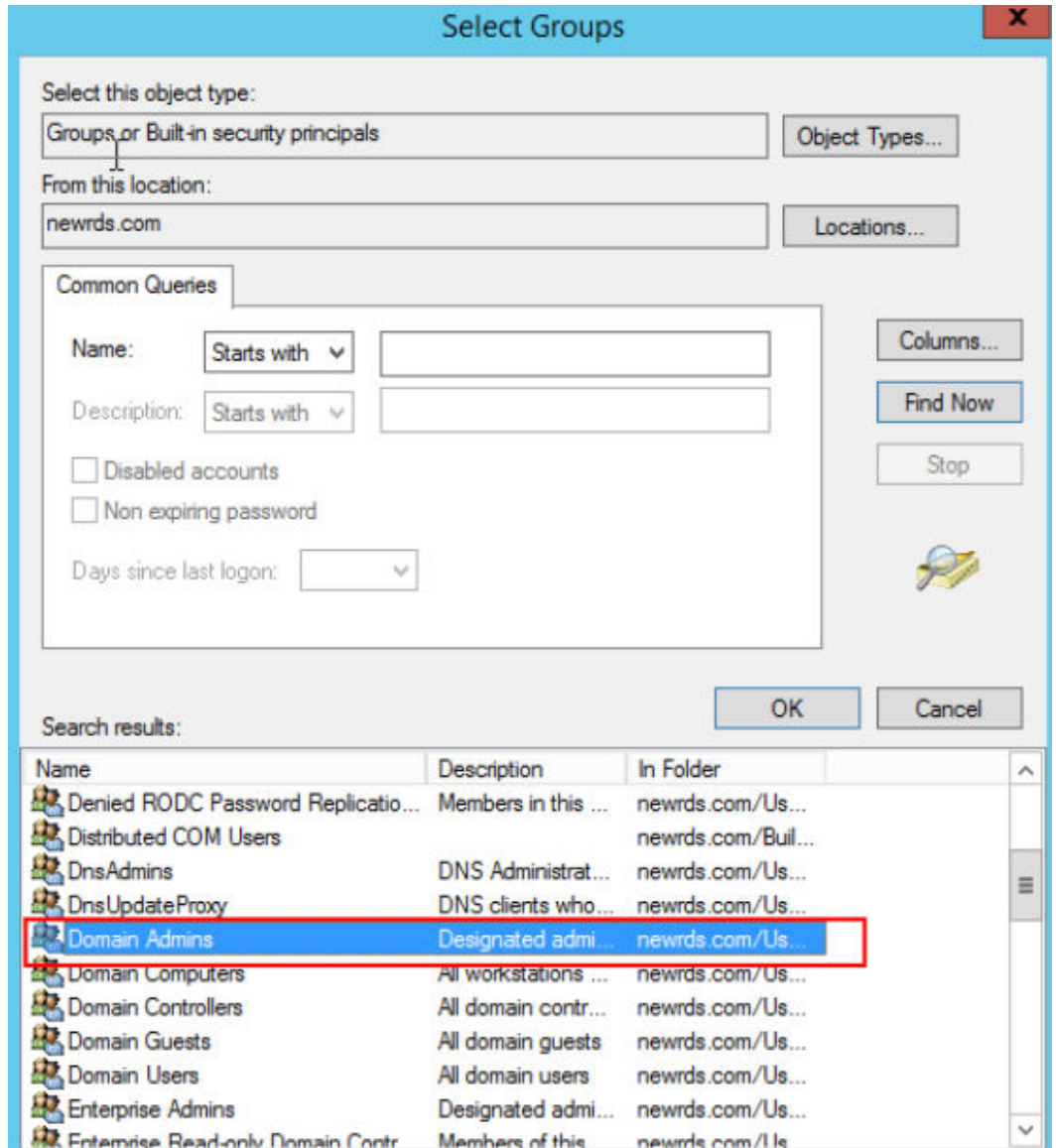
Step 2 Create and add a domain account.

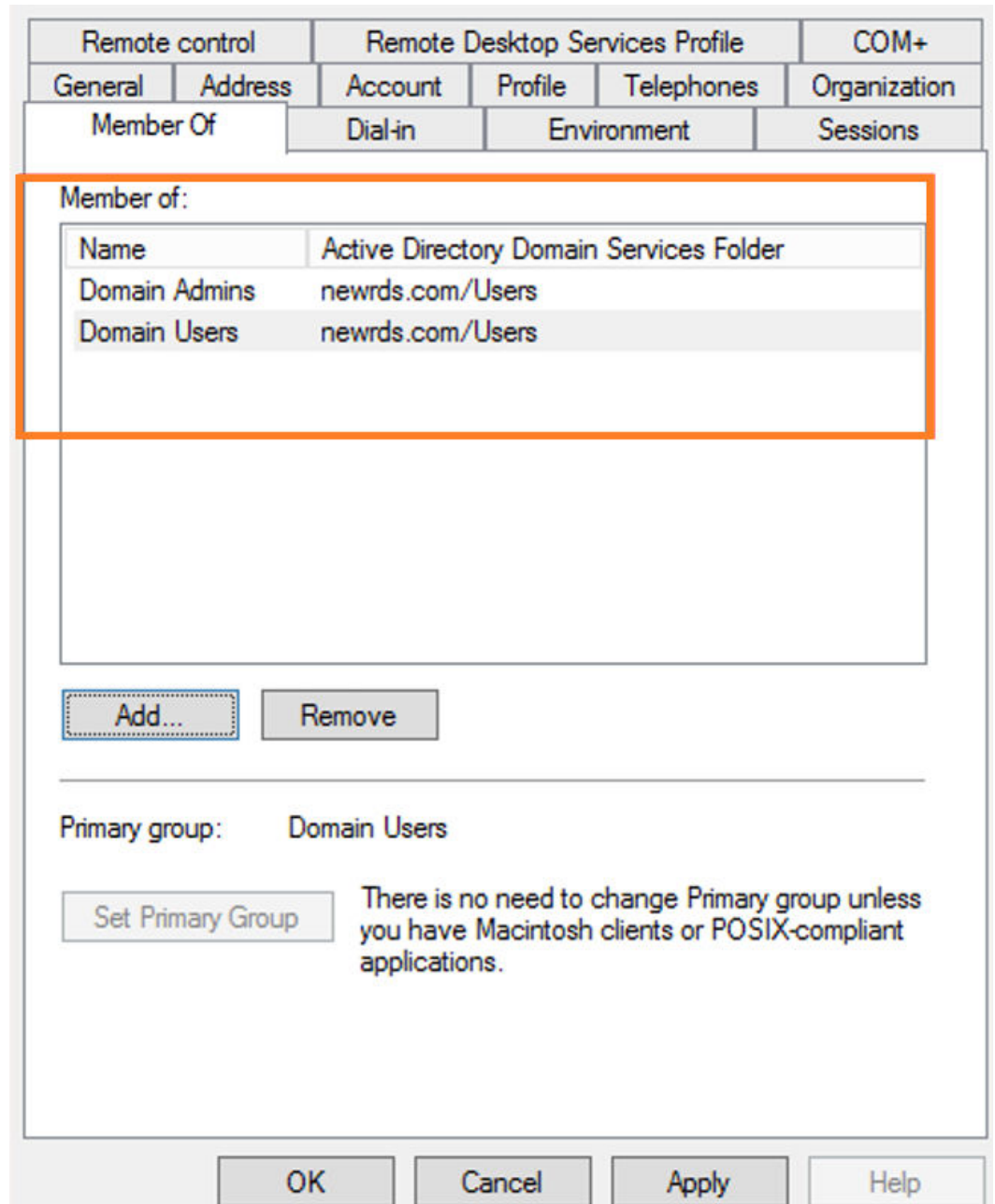
1. Open **Active Directory Users and Computers**, right-click on the **Users** and choose **New > User**. Enter the username and click **Next**.
2. Enter the first name, last name, and user login name, such as luna@newrds.com.
3. Enter the password and then confirm it. Deselect all check boxes (do not change the password at the first login).
4. After the user is added, a figure similar to the following is displayed. You can add domain accounts to user groups for permission control.



5. Add the domain account for logging in to RDS to the Active Directory Admin group.







Step 3 Add an RDS DB instance to the domain.

On the instance creation page, click **Configure** to configure the AD domain and then complete the DB instance creation. After the instance is created, the AD domain can be used.

Table 6-2 AD domain parameters

Parameter	Description
Directory Address	Enter the IP address of the ECS that supports the AD domain. For example: 192.168.x.x. NOTE Ensure that the ECS IP address can connect to the RDS DB instance.
Domain Name	A fully qualified domain name, such as DBStest.com, must: 1. Be the same as the ECS domain name. 2. Be no more than 48 characters long. 3. Only include letters, digits, dots (.), and hyphens (-). 4. Include a valid top-level domain name which contains at least 2 characters long and contains only dots (.) and letters.
Directory Administrator	You are advised to enter the domain administrator username.
Directory Administrator Password	Password of the directory administrator. Keep this password secure. The system cannot retrieve it.

----End

6.1.10 What Can I Do About Websites Responding Slower After Using RDS?

To solve this problem, you are advised to perform the following operations:

- Check the performance status of RDS DB instances on the RDS console.
- Check and compare the current database connection status of the local database and the RDS DB instance. This problem may be related to applications.

6.1.11 How Does a Cloud Database Perform a Primary/Standby Switchover?


RDS provides primary/standby DB instances for high availability. The system will perform a primary/standby failover in case of a failure.


Failover (Automatic)

It is also called out-planned handover. If the primary DB instance fails, the system will automatically switch to the standby DB instance within 5 minutes. No human intervention is required. The connection IP address remains unchanged. DB instances cannot be accessed during the failover. You need to configure automatic reconnections between applications and RDS DB instances to ensure near-continuous availability.

Switchover (Manual)

It is also called out-planned handover. When a DB instance is running properly, you can manually perform a primary/standby switchover as required.

- 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 **Service List**. Under **Database**, click **Relational Database Service**. The RDS console is displayed.
- Step 4** On the **Instance Management** page, click the target DB instance.
- Step 5** In the **DB Information** area on the displayed **Basic Information** page, click **Switch** in the **DB Instance Type** field.


Alternatively, click  in the DB instance topology on the **Basic Information** page to perform a primary/standby switchover.

NOTICE

Primary/standby switchover may cause service interruption for some seconds or minutes (determined by the replication delay). If the primary/standby synchronization delay is too long, a small amount of data may get lost. To prevent traffic congestion, you are advised to perform switchover during off-peak hours.

- Step 6** In the **Switch Primary/Standby Instances** dialog box, click **Yes** to switch between the primary and standby DB instances.

If the replication status is **Available** and the replication delay is greater than 300s, the primary/standby switchover task cannot be delivered.

- Step 7** After a switchover is successful, you can view and manage the DB instance on the **Instance Management** page.
 - During the switchover process, the DB instance status is **Switchover in progress**.
 - In the upper right corner of the DB instance list, click  to refresh the list. After the switchover is successful, the DB instance status will become **Available**

----End

6.1.12 Can Multiple ECSs Connect to the Same RDS DB Instance?

Multiple ECSs can connect to the same RDS DB instance as long as the capability limits of a database are not exceeded.

6.1.13 Can Primary and Standby RDS DB Instances Be Deployed in the Same 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.

RDS allows you to deploy primary/standby DB instances in an AZ or across AZs. You can determine whether the standby AZ is the same as the primary AZ.

- If they are the same (default setting), the primary and standby DB instances are deployed in the same AZ.
- If they are different, the primary and standby DB instances are deployed in different AZs to ensure failover support and high availability.

6.2 Resource and Disk Management

6.2.1 Which Types of Logs and Files Occupy RDS Storage Space?

The following logs and files occupy RDS storage space.

Table 6-3 MySQL database file types

DB Engine	File Type
MySQL	Log files: database undo-log, redo-log, and binlog files
	Data files: database content files and index files
	Other files: ibdata, ib_logfile0, and temporary files

Table 6-4 PostgreSQL database file types

DB Engine	File Type
PostgreSQL	Log files: database error log and transaction log files
	Data files: database content, index, replication slot data, transaction status data, and database configuration files
	Other files: temporary files

Solution

1. If the original storage space is insufficient as your services grow, scale up storage space of your DB instance.

2. If data occupies too much storage space, run **DROP**, **TRUNCATE**, or **DELETE +OPTIMIZE TABLE** to delete useless historical table data to release storage space. If no historical data can be deleted, scale up your storage space.
3. If temporary files generated by sorting queries occupy too much storage space, optimize your SQL query statements.
 - a. A large number of temporary files are generated due to a large number of sorting queries executed by applications.
 - b. A large number of binlog files are generated and occupy space due to large amounts of add, delete, and modify operations in a short period.
 - c. A large number of binlog files are generated due to a large number of transactions and write operations.
4. Use Cloud Eye to monitor the size, usage, and utilization of storage space of your DB instance and set alarm policies.

6.2.2 Which Items Occupy the Storage Space of My RDS DB Instances?

Both your common data (excluding backup data) and the data required for the operation of your DB instances (such as system database data, rollback logs, redo logs, and indexes) occupies the storage space on your purchased RDS DB instances. The following RDS log files also occupy storage space:

- Binlog log files generated by MySQL databases
- Logs files generated by PostgreSQL database servers

These files ensure the stable running of RDS DB instances.

6.2.3 What Overhead Does the Storage Space Have After I Applied for an RDS DB Instance?

The storage space you applied for will contain the system overhead required for inode, reserved block, and database operation.

6.2.4 How Much Storage Space Is Required for DDL Operations?

Data Definition Language (DDL) operations may increase storage space usage sharply. To ensure that services are running properly, do not perform DDL operations during peak hours. If DDL operations are required, ensure that storage space is 10 GB greater than or equal to twice the size of the tablespace. For example, if your tablespace is 500 GB, ensure that storage space is greater than or equal to 1010 GB (500 GB x 2 + 10 GB).

6.2.5 How Many DB Instances Can Run on RDS?

There are no limitations on the number of DB instances running on RDS.

6.2.6 How Many Databases Can Run on an RDS DB Instance?

The maximum number of databases that can run on an RDS DB instance depends on the DB engine settings.

If there are enough CPU, memory, and storage resources, there are no limitations to the number of databases running on a DB instance. A maximum of 0.5 million tables can be backed up. Excessive tables will generate errors. The backup speed is affected by the number of tables in a database.

- MySQL allows you to create numerous databases and tables. For details, see the official MySQL documentation.
- PostgreSQL allows you to create numerous databases and database accounts.

6.3 Database Connection

6.3.1 Can an External Server Access the RDS Database?

DB Instance Bound with an EIP

For a DB instance that has been bound with an EIP, you can access it through the EIP.

DB Instance Not Bound with an EIP

- Enable a VPN in a VPC and use the VPN to connect to the RDS DB instance.
- Create an RDS and an ECS in the same VPC and access RDS through the ECS.

6.3.2 How Do I Troubleshoot If the Number of RDS Database Connections Reaches the Upper Limit?

The number of database connections indicates the number of applications that can be simultaneously connected to a database, and is irrelevant to the maximum number of users allowed by your applications or websites.

If there is an excessive number of database connections, applications may fail to be connected, and the full and incremental backups may fail, affecting service running.

Fault Locating

1. Check whether applications are connected, optimize the connections, and release unnecessary connections.
2. Check whether the specifications are small and scale them as needed.
3. Check whether any metrics are abnormal and whether any alarms are generated on the Cloud Eye console. Cloud Eye monitors database metrics, such as the CPU usage, memory usage, storage space usage, and database connections, and allows you to set alarm policies to identify risks in advance if any alarms are generated. For details, see the *Cloud Eye User Guide*.

6.3.3 What Is the Maximum Number of Connections to an RDS DB Instance?

RDS does not have constraints on the number of connections. This number is determined by the default value and value range of the DB engine. For example,

you can set **max_connections** and **max_user_connections** in a parameter template to configure the maximum number of connections for an RDS MySQL DB instance.

About max_connections

The max_connections is closely related to storage space (unit: GB) of the DB instance.

Estimated max_connections = Available node memory/Estimated memory occupied by a single connection

NOTE

- Available node memory = Total memory – Memory occupied by the buffer pool – 1 GB (mysqld process/OS/monitoring program)
- Estimated memory usage of a single connection (single_thread_memory) = thread_stack (256 KB) + binlog_cache_size (32 KB) + join_buffer_size (256 KB) + sort_buffer_size (256 KB) + read_buffer_size (128 KB) + read_rnd_buffer_size (256 KB) ≈ 1 MB

The following table lists the default values of **max_connections** for different memory specifications.

Table 6-5 Max_connections for different memory specifications

Memory (GB)	Connections
512	100,000
384	80,000
256	60,000
128	30,000
64	18,000
32	10,000
16	5,000
8	2,500
4	1,500
2	800

6.3.4 How Can I Create and Connect to an ECS?

1. For details about how to create an ECS, see the *Elastic Cloud Server User Guide*.
 - The ECS is used for connecting to an RDS DB instance and must be located in the same VPC as the RDS DB instance.
 - Configure a correct security group to allow the ECS to access the RDS DB instance through the private address.

2. For details on how to connect to the ECS, see the "Logging in to an ECS" section in the *Elastic Cloud Server User Guide*.

6.3.5 What Should I Do If an ECS Cannot Connect to an RDS DB Instance?

Perform the following steps to identify the problem:

- Step 1** Check whether the ECS and RDS DB instance are located in the same VPC.
- If they are in the same VPC, go to [Step 2](#).
 - If they are in different VPCs, create an ECS in the VPC in which the RDS DB instance is located.
- Step 2** Check whether a security group has been added to the ECS.
- If no security group has been added, go to the VPC console from the ECS details page and click **Security Groups** to add a security group.
- Step 3** On the ECS, check whether the RDS DB instance port can be connected.

The default port of RDS for MySQL is **3306**.

The default port of RDS for PostgreSQL is **5432**.

```
telnet <IP address> {port number}
```

- If the ECS can connect to the RDS DB instance port, the network between the ECS and the RDS DB instance is normal.
- If the ECS cannot connect to the port, contact technical support.

----End

6.3.6 What Should I Do If a Database Client Problem Causes a Connection Failure?

Identify an RDS connection failure caused by a client problem from the following aspects.

1. ECS Security Policy
In Windows, check whether the RDS instance port is enabled in the Windows security policy. In Linux, run the **iptables** command to check whether the RDS DB instance port is enabled in firewall settings.
2. Application Configuration
Check whether the connection address, port parameter configuration, and JDBC connection parameter configuration are correct.
3. Incorrect User Name or Password
Check whether the user name or password is correct if an error similar to the following occurs during RDS DB connection:
 - [Warning] Access denied for user 'username'@'yourIp' (using password: NO)
 - [Warning] Access denied for user 'username'@'yourIp' (using password: YES)

 NOTE

If the problem persists, contact post-sales technical support.

6.3.7 What Should I Do If an RDS Database Problem Causes a Connection Failure?

Check whether any of the following problems occur on the RDS DB instance.

1. The RDS DB instance is not properly connected.
Solution: Check the connection. The RDS DB instance must be accessed only through an ECS in the same VPC.
2. The maximum number of connections has been reached.
Solution: Check whether the CPU usage and the number of current connections are normal by using the RDS resource monitoring function. If either of them reaches the maximum, reboot, disconnect, or scale up the specifications of the DB instance.
3. DB instance is abnormal. For example, the RDS DB instance fails to be rebooted, the system is faulty, or the instance or table is locked.
Solution: Reboot the RDS DB instance to see if the problem is resolved. If the problem persists, contact post-sales technical support.

6.3.8 How Do My Applications Access an RDS DB Instance in a VPC?

Ensure that the ECS in which your applications are located is in the same VPC as the RDS DB instance. If the ECS and the RDS DB instance are in different VPCs, modify the VPC route table and network access control list (ACL) to ensure that the ECS can access the RDS DB instance.

6.3.9 Do Applications Need to Support Reconnecting to the RDS DB Instance Automatically?

It is recommended that your applications support automatic reconnections to the database. After a database reboot, your applications will automatically reconnect to the database to increase service availability and continuity.

In addition, you are advised to set your applications to connect to the database using a long connection to reduce resource consumption and improve performance.

6.3.10 How Can I Connect to a PostgreSQL Database Through JDBC?

If you are connecting to a PostgreSQL database through Java database connectivity (JDBC), the SSL certificate is optional. For security reasons, you are advised to download the SSL certificate to encrypt the connection.

Prerequisites


You must be familiar with:

- Computer basics
- Java programming language
- JDBC basic knowledge

Connection with the SSL Certificate

NOTE

The JDBC connection is an SSL connection. The SSL certificate needs to be downloaded and verified for connecting to databases.

In the **DB Information** area on the **Basic Information** page, click  in the **SSL** field to download the root certificate or certificate bundle.

Step 1 Connect to the RDS PostgreSQL DB instance through JDBC.

```
jdbc:postgresql://<instance_ip>:<instance_port>|<database_name>?sslmode=verify-
full&sslrootcert=<ca.pem>
```

Table 6-6 Parameter description

Parameter	Description
<instance_ip>	If you are accessing the RDS DB instance through an ECS, instance_ip indicates the floating IP address displayed on the Basic Information page of the DB instance to which you intend to connect. If you are accessing the RDS DB instance through an EIP, instance_ip indicates the EIP that has been bound to the DB instance.
<instance_port>	Indicates the database port number displayed on the Basic Information page. The default port number is 5432 .
<database_name>	Indicates the name of the database to which you intend to connect. The default database name is postgres .
sslmode	Indicates the SSL connection mode. The default mode is verify-full .
sslrootcert	Indicates the directory of the CA certificate for the SSL connection. The certificate should be stored in the directory where the command is executed.

Example script in Java:

```
import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.ResultSet;
import java.sql.Statement;

public class MyConnTest {
    final public static void main(String[] args) {
        Connection conn = null;
        // set sslmode here.
        // with ssl certificate and path.
```

```
String url = "jdbc:postgresql://192.168.0.225:5432/my_db_test?sslmode=verify-
full&sslrootcert=/home/Ruby/ca.pem";

try {
    Class.forName("org.postgresql.Driver");
    conn = DriverManager.getConnection(url, "root", "password");
    System.out.println("Database connected");

    Statement stmt = conn.createStatement();
    ResultSet rs = stmt.executeQuery("SELECT * FROM mytable WHERE columnfoo = 500");
    while (rs.next()) {
        System.out.println(rs.getString(1));
    }

    rs.close();
    stmt.close();
    conn.close();
} catch (Exception e) {
    e.printStackTrace();
    System.out.println("Test failed");
} finally {
    // release resource ....
}
}
```

----End

Connection Without the SSL Certificate

NOTE

The JDBC connection is an SSL connection, but you do not need to download the SSL certificate because the certificate verification on the server is not required.

Step 1 Connect to the RDS PostgreSQL DB instance through JDBC.

```
jdbc:postgresql://<instance_ip>:<instance_port>/<database_name>?sslmode=disable
```

Table 6-7 Parameter description

Parameter	Description
<instance_ip>	If you are accessing the RDS DB instance through an ECS, instance_ip indicates the floating IP address displayed on the Basic Information page of the DB instance to which you intend to connect. If you are accessing the RDS DB instance through an EIP, instance_ip indicates the EIP that has been bound to the DB instance.
<instance_port>	Indicates the database port number displayed on the Basic Information page. The default port number is 5432 .
<database_name>	Indicates the name of the database to which you intend to connect. The default database name is postgres .
sslmode	Indicates the SSL connection mode. disable indicates that data is not encrypted.

Example script in Java:

```
import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.ResultSet;
import java.sql.Statement;

public class MyConnTest {
    final public static void main(String[] args) {
        Connection conn = null;
        // set sslmode here.
        // no ssl certificate, so do not specify path.
        String url = "jdbc:postgresql://192.168.0.225:5432/my_db_test?sslmode=disable";
        try {
            Class.forName("org.postgresql.Driver");
            conn = DriverManager.getConnection(url, "root", "password");
            System.out.println("Database connected");

            Statement stmt = conn.createStatement();
            ResultSet rs = stmt.executeQuery("SELECT * FROM mytable WHERE columnfoo = 500");
            while (rs.next()) {
                System.out.println(rs.getString(1));
            }
            rs.close();
            stmt.close();
            conn.close();
        } catch (Exception e) {
            e.printStackTrace();
            System.out.println("Test failed");
        } finally {
            // release resource ....
        }
    }
}
```

----End

6.3.11 Can I Access an RDS DB Instance Over an Intranet Across Regions?

Sorry, but you cannot. RDS DB instances in different regions cannot communicate with each other over an intranet. For low network latency and quick resource access, select the nearest region.

6.3.12 Is an SSL Connection to a DB Instance Interrupted After a Primary/Standby Switchover or Failover Occurs?

For DB instances connected through SSL, a primary/standby switchover or failover does not interrupt the connection because the SSL certificate is still valid for both the primary and standby DB instances.

6.3.13 Does MySQL Support SSL Connections?

MySQL supports SSL connections. Different from other vendors, RDS for MySQL enables the SSL connection on the database server by default. When you use a

client to connect to MySQL DB instances, you can determine whether to enable SSL as required.

For details about SSL, see section [Connecting to a MySQL DB Instance Through a Public Network](#).

6.3.14 Why Does the New Password Not Take Effect After I Reset the Administrator Password?

Possible Causes

You reset the administrator password after the backup is created. Therefore, the original administrator password takes effect after data is restored from the backup.

Locating Method

Check whether the DB instance was restored after you reset the administrator password.

Solution

Log in to the RDS console and reset the administrator password again. For details, see section [Resetting the Administrator Password](#).

6.4 Database Migration

6.4.1 Why Do I Need to Use the mysqldump and pg_dump Tools for Migration?

The mysqldump or pg_dump tool is easy to use for data migration. However, when you use this tool, the server is stopped for a long period of time during data migration. Therefore, use these tools when the data amount is small or if the server is allowed to stop for a long period of time, during which the data can be migrated.

RDS is compatible with original database services. The procedure for migrating data from your database to RDS is similar to the procedure for migrating data from one database server to another.

6.4.2 What Types of DB Engines Does RDS Support for Importing Data?

- Exporting or importing data between DB engines of the same type is called homogeneous database export or import.
- Exporting or importing data between DB engines of different types is called heterogeneous database export or import. For example, import data from Oracle to DB engines supported by RDS.

Data cannot be exported or imported between heterogeneous databases due to different data formats. However, if the data formats are compatible, table data can also be imported theoretically.

Generally, third-party software is required for data replication to export and import between heterogeneous databases. For example, you can use a third-party tool to export table records from Oracle in .txt format. Then, you can use Load statements to import the exported table records to the DB engines supported by RDS.

6.5 Database Permission

6.5.1 Why Does the Root User Not Have the Super Permission?

Most relational database cloud service platforms do not provide the super permission for the **root** user. The super permission allows users to execute many management commands, such as reset master, set global, kill, and reset slave. These operations may cause primary/standby replication errors. To ensure stable running of DB instances, RDS does not provide the super permission for the **root** user.

If you require the super permission, RDS can provide service capabilities or use other methods to bypass the super permission constraints.

For example:

1. You cannot run the following command on a database to modify parameter values. You can modify parameter values only on the RDS console.

set global parameter name=*Parameter value*;

If the script contains the **set global** command and causes the super permission loss, delete the **set global** command and modify parameter values through the RDS console.

2. You can create PostgreSQL plugins by referring to [Creating and Deleting a Plugin](#).

6.6 Database Storage

6.6.1 What Storage Engines Does the RDS for MySQL Support?

Database storage engine is a core service for **storing, processing, and protecting data**. It can be used to control access permissions and rapidly process transactions to meet enterprise requirements.

For MySQL databases, only the InnoDB storage engine supports backup and restoration functions and is therefore recommended.

For versions later than MySQL 5.6.40 and 5.7.22, some storage engines are no longer supported.

RDS for MySQL now does not support MyISAM due to the following reasons:

- MyISAM engine tables do not support transactions and support only table-level locks. As a result, read and write operations conflict with each other.
- MyISAM has a defect in protecting data integrity, which may cause database data damage or even data loss.
- If data is damaged, MyISAM does not support data restoration provided by RDS for MySQL and requires manual restoration.
- Data can be transparently migrated from MyISAM to InnoDB, which does not require code modification for tables.

RDS for MySQL now does not support FEDERATED due to the following reasons:

- Same DML operations are repeatedly executed on remote databases, causing data disorder.
- During the PITR restoration, data on remote databases is not restored to the status when the full backup is created after the full restoration phase is complete. Applying data during the incremental restoration will disorder FEDERATED table data.

RDS for MySQL now does not support MEMORY due to the following reasons:

- If a memory table becomes empty after a restart, the database generates a DELETE event to the binlog when the table is opened. If primary/standby DB instances use memory tables and the standby database (or read-only database) is restarted, a GTID is generated, which is inconsistent with that of the primary database. As a result, the standby database is rebuilt.
- Using memory tables may cause out-of-memory (OOM) and even service termination.

6.6.2 What Is the RDS DB Instance Storage Configuration?

RDS uses EVS disks for data storage. For EVS details, see *Elastic Volume Service User Guide*.

The RDS DB instance backup data is stored in OBS and does not occupy the database storage space. For details on the RDS DB instance storage configuration, see the *Object Storage Service User Guide*.

6.6.3 Can I Change the Storage Type of an RDS DB Instance from Common I/O to Ultra-high I/O?

No. After an RDS DB instance is created, the storage type cannot be changed.

Table 6-8 Items that cannot be changed

Item	Change Direction
Storage type	<ul style="list-style-type: none"> • From common I/O to ultra-high I/O • From ultra-high I/O to common I/O • From high I/O to common I/O <p>The preceding descriptions are examples only. The storage type cannot be changed.</p>

6.6.4 What Should I Do If My Data Exceeds the Database Storage Space of an RDS DB Instance?

Scenario

The database storage space of an RDS DB instance is exhausted, and applications cannot read data from or write data to databases, interrupting services.

Cause

1. Data occupies a great amount of storage space.
2. A large number of binlog files are generated due to a large number of transactions and write operations.
3. A large number of temporary files are generated due to a large number of sorting queries executed by applications.

Solution

1. If data occupies too much storage space, run **DROP**, **TRUNCATE**, or **DELETE +OPTIMIZE TABLE** to delete useless historical table data to release storage space. If no historical data can be deleted, scale up your storage space.
2. If binlog files occupy too much storage space, contact technical support to delete local binlog files to release storage space.
3. If temporary files generated by sorting queries occupy too much storage space, optimize your SQL query statements.
4. If the preceding solutions are invalid, implement database and table sharding.

6.7 Client Installation

6.7.1 How Can I Install the MySQL Client?

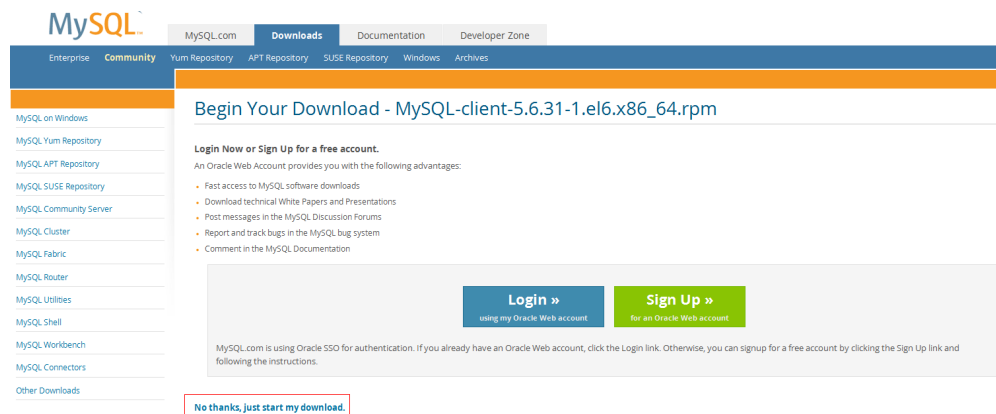
MySQL provides client installation packages for different OSs on its official website. MySQL 5.6 is used as an example. Click [here](#) to download the MySQL 5.6 client installation package or click [here](#) to download other versions of the packages. The following procedure uses Red Hat Linux OS as an example to illustrate how to obtain the required installation package and install the MySQL client.

Procedure

Step 1 Obtain the installation package.

Find the link to the required version on the download page. MySQL-client-5.6.31-1.el6.x86_64.rpm is used as an example in the following figure.

Figure 6-1 Procedure



NOTE

Click **No thanks, just start my download.** to download the installation package.

Step 2 Upload the installation package to the ECS.

NOTE

When you create an ECS, select an OS, such as Red Hat 6.6, and bind an EIP to it. Then, upload the installation package to the ECS using a remote connection tool, and use PuTTY to connect to the ECS.

Step 3 Run the following command to install the MySQL client:

```
sudo rpm -ivh MySQL-client-5.6.31-1.el6.x86_64.rpm
```

NOTE

- If any conflicts occur during the installation, add the **replacefiles** parameter to the command and try to install the client again. Example:
rpm -ivh --replacefiles MySQL-client-5.6.31-1.el6.x86_64.rpm
- If a message is displayed prompting you to install a dependency package, you can add the **nodeps** parameter to the command and install the client again. Example:
rpm -ivh --nodeps MySQL-client-5.6.31-1.el6.x86_64.rpm

----End

6.7.2 How Can I Install the PostgreSQL Client?

PostgreSQL provides **client installation methods** for different OSs on its official website.

The following describes how to install a PostgreSQL 12 client in CentOS.

Procedure

Step 1 Log in to an ECS.

1. When you create an ECS, select an OS like CentOS 7 and bind an EIP to it.
2. Use a remote connection tool to connect to the ECS through the EIP.

Step 2 Open the **client installation page**.

Step 3 Select a DB engine version, OS, and OS architecture, and run the following commands on the ECS to install a PostgreSQL client.


```
sudo yum install -y https://download.postgresql.org/pub/repos/yum/reporepms/EL-7-x86_64/pgdg-redhat-repo-latest.noarch.rpm
sudo yum install -y postgresql12-server
```

Figure 6-2 Installing a client

To use the PostgreSQL Yum Repository, follow these steps:

- Select version:
- Select platform:
- Select architecture:
- Copy, paste and run the relevant parts of the setup script:


```
# Install the repository RPM:
sudo yum install -y https://download.postgresql.org/pub/repos/yum/reporepms/EL-7-x86_64/pgdg-redhat-repo-latest.noarch.rpm

# Install PostgreSQL:
sudo yum install -y postgresql12-server

# Optionally initialize the database and enable automatic start:
sudo /usr/pgsql-12/bin/postgresql-12-setup initdb
sudo systemctl enable postgresql-12
sudo systemctl start postgresql-12
```

Copy Script

- Select a DB engine version that is consistent with that of your RDS for PostgreSQL instance.
- Select an OS that is consistent with that of the ECS.
- Select an OS architecture that is consistent with that of the ECS.

Figure 6-3 Installing the RPM package

```
root@ecs-d605 ~]# sudo yum install -y https://download.postgresql.org/pub/repos/yum/reporepms/EL-7-x86_64/pgdg-redhat-repo-latest.noarch.rpm
Loaded plugins: fastestmirror
pgdg-redhat-repo-latest.noarch.rpm | 8.6 kB 00:00:00
Examining /var/tmp/yum-root-2onITG/pgdg-redhat-repo-latest.noarch.rpm: pgdg-redhat-repo-42.0-28.noarch
Marking /var/tmp/yum-root-2onITG/pgdg-redhat-repo-latest.noarch.rpm to be installed
Resolving Dependencies
--> Running transaction check
--> Package pgdg-redhat-repo.noarch 0:42.0-28 will be installed
--> Finished Dependency Resolution

Dependencies Resolved

=====
Package Arch Version Repository Size
=====
Installing:
pgdg-redhat-repo noarch 42.0-28 /pgdg-redhat-repo-latest.noarch 13 k
Transaction Summary
-----
Install 1 Package

Total size: 13 k
Installed size: 13 k
Downloading packages:
Running transaction check
Running transaction test
Transaction test succeeded
Running transaction
Warning: RPMDB altered outside of yum.
Installing : pgdg-redhat-repo-42.0-28.noarch 1/1
Verifying : pgdg-redhat-repo-42.0-28.noarch 1/1

Installed:
pgdg-redhat-repo.noarch 0:42.0-28
Complete!
```

Figure 6-4 Client installed

```

Total                                                                 467 kB/s | 14 MB  00:00:30
Retrieving key from file:///etc/pki/rpm-gpg/RPM-GPG-KEY-PGDG
Importing GPG key 0x442DF0F8:
  Userid   : "PostgreSQL RPM Building Project <pgsql-pkg-june@postgresql.org>"
  Fingerprint: 60c9 e2b9 1a37 d136 fe74 d176 1f16 d2e1 442d f0f8
  Package   : pgdg-redhat-repo-42.0-20.noarch (@/pgdg-redhat-repo-latest.noarch)
  From      : /etc/pki/rpm-gpg/RPM-GPG-KEY-PGDG
Running transaction check
Running transaction test
Transaction test succeeded
Running transaction
  Installing : libicu-58.2-4.e17_7.x86_64                                1/4
  Installing : postgresql12-libs-12.13-1PGDG.rhel7.x86_64            2/4
  Installing : postgresql12-12.13-1PGDG.rhel7.x86_64                3/4
  Installing : postgresql12-server-12.13-1PGDG.rhel7.x86_64         4/4
  Verifying  : postgresql12-libs-12.13-1PGDG.rhel7.x86_64          1/4
  Verifying  : postgresql12-12.13-1PGDG.rhel7.x86_64              2/4
  Verifying  : postgresql12-server-12.13-1PGDG.rhel7.x86_64       3/4
  Verifying  : libicu-58.2-4.e17_7.x86_64                          4/4

Installed:
  postgresql12-server.x86_64 0:12.13-1PGDG.rhel7

Dependency Installed:
  libicu.x86_64 0:58.2-4.e17_7      postgresql12.x86_64 0:12.13-1PGDG.rhel7      postgresql12-libs.x86_64 0:12.13-1PGDG.rhel7

Complete!

```

Step 4 Connect to the RDS for PostgreSQL instance.

Figure 6-5 Connection successful

```

[root@ecs-d605 ~]# psql -h [redacted] -d postgres -U root
Password for user root:
psql (12.13, server 12.11)
SSL connection (protocol: TLSv1.2, cipher: ECDHE-RSA-AES256-GCM-SHA384, bits: 256, compression: off)
Type "help" for help.

postgres=>

```

----End

6.8 Backup and Restoration

6.8.1 How Long Does RDS Store Backup Data?

Automated backup data is kept based on the backup retention period you specified. There is no limit on the manual backup retention period. You can delete manually backup files as needed.

The backup data is stored in OBS and does not occupy the database storage space.

6.8.2 Can My Database Be Used in the Backup Window?

A backup window is a user-specified time segment during which backup of RDS DB instances is performed. With these periodic data backups, RDS allows you to restore DB instances to the backups during a retention period. This backup process does not affect services. However, you cannot reboot DB instances on the RDS console.

6.8.3 How Can I Back Up RDS Databases to an ECS?

You can back up data to an ECS the same way you export SQL statements. The ECS service does not have restrictions on the types of data to be backed up as long as the data complies with local laws and regulations. You can store RDS backup data on an ECS. However, you are advised not to use an ECS as the

database backup space. You are advised to store RDS backup data to OBS for high data reliability and service assurance.

6.8.4 Why Has My Automated Backup Failed?

Automated backup failures may be caused by the following reasons:

1. The network environment is unstable, due to issues such as network delay or interruption. RDS will detect these problems and trigger an automated backup after half an hour. You can also perform a manual backup before then.
2. Multi-task executions are complicated, resulting in problems such as task waiting or interruption. RDS will detect these problems and trigger an automated backup half an hour later. You can also perform a manual backup in time.
3. A DB instance status is unavailable, possibly because the DB instance is faulty or being modified. RDS will trigger an automated backup after the DB instance status becomes available. You can also perform a manual backup before then.
4. A parameter change is incorrect. For example, a DB instance may be faulty after a parameter template containing incorrectly changed parameters apply to it. You can check whether original and current values are correct, check whether any related parameters also need to be changed, reset the parameter template, or reboot the DB instance.
5. An error has occurred during data import.
For example, system table records get lost due to inappropriate data import.
If the problem persists, contact technical support.

6.8.5 What Happens to Database Backups After an RDS DB Instance Is Deleted?

When you delete a DB instance, its automated backups are also deleted but its manual backups are retained.

6.8.6 Will My Backups Be Deleted If I Delete My Cloud Account?

If your cloud account is deleted, both your automated and manual backups are deleted.

6.8.7 Why Is a Table or Data Missing from My Database?

RDS does not delete or perform any operations on any user data. If this problem occurs, check whether a misoperation has been performed. Restore the data using backup files, if necessary.

Possible solutions are as follows:

- Use the RDS restoration function to restore data.
- Import the backup data to RDS through an ECS.

6.9 Database Monitoring

6.9.1 Which DB Instance Monitoring Metrics Do I Need to Pay Most Attention To?

You need to pay the most attention to CPU, memory, and storage space usage.

To stay aware of these metrics, you can configure the system to report alarms to Cloud Eye as needed. You can then take measures to clear any reported alarms.

Configuration examples:

- Configure RDS to report alarms to Cloud Eye if its CPU utilization reaches or exceeds a specific value (for example, 90%) multiple times (for example, 3 times) within a set period (for example, 5 minutes).
- Configure RDS to report alarms to Cloud Eye if its memory utilization reaches or exceeds a specific value (for example, 90%) multiple times (for example, 4 times) within a set period (for example, 5 minutes).
- Configure RDS to report alarms to Cloud Eye if its storage utilization reaches or exceeds a specific value (for example, 85%) multiple times (for example, 5 times) within a set period (for example, 5 minutes).

NOTE

For details on Cloud Eye alarm configuration, see the "Alarm Rule Management" section in the *Cloud Eye User Guide*.

Measures:

- If a CPU or memory alarm is reported, you can scale up the vCPUs or memory by changing the DB instance class.
- If a storage space usage alarm is reported, perform either of the following operations:
 - Check the storage space consumption to see whether any space can be freed up by deleting data from DB instances or dumping the data to another system.
 - Scale up the storage space.

6.10 Capacity Expansion and Specification Change

6.10.1 Are My RDS DB Instances Available When Scaling?

Currently, you can scale up storage space and change the CPU or memory of a DB instance.

- When scaling storage space, RDS DB instances are available and services are not affected. However, you cannot delete or reboot DB instances that are being scaled.
- When changing the CPU or memory of DB instances, the network is intermittently disconnected for one or two times within seconds. For primary/

standby DB instances, a failover may occur and services may be interrupted for a short period of time.

6.10.2 Why Does the DB Instance Become Faulty After the Original Database Port Is Changed?

Symptom

- The DB instance is in **Faulty** state after the original database port is changed.
- The DB instance cannot be connected using the new database port.

Possible Causes

The submitted database port is occupied.

Procedure

- If the original database port is changed successfully, the previous change failed because the submitted database port is occupied.
- If the original database port still fails to be changed, contact technical support.

6.11 Database Parameter Modification

6.11.1 What Inappropriate Parameter Settings Cause Unavailability of the PostgreSQL Database?

In the following cases, inappropriate parameter settings cause unavailability of the database:

- Parameter value ranges are related to DB instance specifications.
The maximum values of **shared_buffers** and **max_connections** are related to the DB instance physical memory. If you set the parameters inappropriately, the database is unavailable.
- Parameter association is incorrect.
 - If **log_parser_stats**, **log_planner_stats**, or **log_executor_stats** is enabled, you must disable **log_statement_stats**. Otherwise, the database is unavailable.
 - **max_connections**, **autovacuum_max_workers**, and **max_worker_processes** must meet the following requirements. Otherwise, the database is unavailable.
 $\text{max_connections value} + \text{autovacuum_max_workers value} + \text{max_worker_processes value} + 1 < 8388607$

NOTE

For details on parameter descriptions, visit the [PostgreSQL official website](#).

Solution:

1. Log in to the RDS console and query the logs to locate the incorrectly configured parameter.
2. On the **Configuration** page, change parameters to default values and reboot the database.
3. Set the incorrectly configured parameter to a correct value and other parameters to the original values.

6.12 Log Management

6.12.1 How Long Is the Delay of RDS MySQL Slow Query Logs?

Generally, the delay is 5 minutes. If the size of slow query logs reaches 10 MB within 5 minutes, the logs will be uploaded to OBS.

6.12.2 Can I Export Statistics on RDS Slow Query Logs?

Sorry, statistics on RDS slow query logs cannot be exported.

6.13 Network Security

6.13.1 What Security Protection Policies Does RDS Have?

Network

- RDS runs your DB instances in a VPC, ensuring that the DB instances are isolated from other services.
- RDS uses security groups to ensure that only trusted sources can access your DB instances.
- RDS supports SSL connections to encrypt data during transmission.

Management

You can use the Identity and Access Management (IAM) service to manage RDS permissions.

6.13.2 How Can I Ensure the Security of RDS DB Instances in a VPC?

The VPC security group helps ensure the security of RDS in a VPC. In addition, ACL can be used to allow or reject I/O network traffic for each subnet.

6.13.3 How Can Data Security Be Ensured During Transmission When I Access RDS Through an EIP?

When you access RDS through an EIP, service data will be transmitted on the public network. To prevent data breach, you are advised to use SSL to encrypt

data transmitted on the public network. You can also use the Direct Connect or VPN services to encrypt data transmission channels.

6.13.4 How Can I Prevent Untrusted Source IP Addresses from Accessing RDS?

- After you enable public accessibility, your EIP DNS and database port may be obtained by malicious personnel. To protect your information including your EIP, DNS, database port, database account, and password, you are advised to set the range of source IP addresses in the RDS security group to ensure that only trusted source IP addresses can access your DB instances.
- To prevent your database password from being maliciously cracked, set a strong password according to the password strength policies and periodically change it.

6.13.5 How Can I Import the Root Certificate to the Windows or Linux OS?

Importing the Root Certificate to the Windows OS

1. Click **Start** and choose **Run**. In the displayed **Run** dialog box, enter **MMC** and press **Enter**.
2. On the displayed console, choose **File > Add/Remove Snap-in**.
3. In the left **Available snap-ins** pane of the displayed **Add or Remove Snap-ins** dialog box, select **Certificates** and click **Add**.
4. In the displayed **Certificates snap-in** dialog box, select **Computer account** and click **Next**.
5. In the displayed **Select Computer** dialog box, click **Finish**.
6. In the **Add or Remove Snap-ins** dialog box, click **OK**.
7. On the console, double-click **Certificates**.
8. Right-click **Trusted Root Certification Authorities** and choose **All Tasks > Import**.
9. In the displayed **Certificate Import Wizard** dialog box, click **Next**.
10. Click **Browse** to change the file type to **All Files (*.*)**.
11. Locate the downloaded root certificate ca.pem file and click **Open**. Then, click **Next**.

NOTICE

You must change the file type to **All Files (*.*)** because **.pem** is not a standard certificate extension name.

12. Click **Next**.
13. Click **Finish**.
14. Click **OK** to complete the import of the root certificate.

Importing the Root Certificate to the Linux OS


You can use a connection tool (such as WinSCP or PuTTY) to upload the certificate to any directory of the Linux OS.

6.13.6 How Can I Identify the Validity Period of an SSL Root Certificate?

When you connect to an RDS MySQL DB instance using an SSL connection, run the following command to check whether the certificate has expired:

```
show status like '%ssl_server%';
```

Update the root certificate to the latest version before it expires:

1. In the **DB Information** area on the **Basic Information** page, click  in the **SSL** field to download the new root certificate or certificate bundle.
2. Reboot the DB instance for the new certificate to take effect.
3. Connect to the DB instance using the new certificate or certificate bundle.

NOTE

If a certificate is about to expire, replace it with an officially issued certificate to improve system security.

6.13.7 How Can I Identify Data Corruption?

- **Data tempering**
Lots of security measures are provided to ensure that only authenticated users have permissions to perform operations on database table records. The SSH protocol is inaccessible to users. Database tables can be accessed only through the specified database service port.
Verifying package during primary/standby synchronization can prevent data tampering. MySQL uses the InnoDB storage engine to prevent data damage.
- **DB instance servers may be powered off suddenly, causing database page corruption and database rebooting failures.**
If the primary DB instance is faulty, RDS switches to the standby DB instance within 1 to 5 minutes to provide services for you. Databases cannot be accessed during failover. You must set automatic reconnection between your applications and RDS to prevent your applications from becoming unavailable after the failover.