ALIBABA CLOUD

Alibaba Cloud

Apsara Stack Enterprise

ApsaraDB for RDS

ApsaraDB RDS for SQL Server User Guide

Product Version: v3.16.2

Document Version: 20220913

Alibaba Cloud

Legal disclaimer

Alibaba Cloud reminds you to carefully read and fully understand the terms and conditions of this legal disclaimer before you read or use this document. If you have read or used this document, it shall be deemed as your total acceptance of this legal disclaimer.

1. You shall download and obtain this document from the Alibaba Cloud website or other Alibaba Cloud-authorized channels, and use this document for your own legal business activities only. The content of this document is considered confidential information of Alibaba Cloud. You shall strictly abide by the confidentiality obligations. No part of this document shall be disclosed or provided to any third party for use without the prior written consent of Alibaba Cloud.

2. No part of this document shall be excerpted, translated, reproduced, transmitted, or disseminated by any organization, company or individual in any form or by any means without the prior written consent of Alibaba Cloud.

3. The content of this document may be changed because of product version upgrade, adjustment, or other reasons. Alibaba Cloud reserves the right to modify the content of this document without notice and an updated version of this document will be released through Alibaba Cloud-authorized channels from time to time. You should pay attention to the version changes of this document as they occur and download and obtain the most up-to-date version of this document from Alibaba Cloud-authorized channels.

4. This document serves only as a reference guide for your use of Alibaba Cloud products and services. Alibaba Cloud provides this document based on the "status quo", "being defective", and "existing functions" of its products and services. Alibaba Cloud makes every effort to provide relevant operational guidance based on existing technologies. However, Alibaba Cloud hereby makes a clear statement that it in no way guarantees the accuracy, integrity, applicability, and reliability of the content of this document, either explicitly or implicitly. Alibaba Cloud shall not take legal responsibility for any errors or lost profits incurred by any organization, company, or individual arising from download, use, or trust in this document. Alibaba Cloud shall not, under any circumstances, take responsibility for any indirect, consequential, punitive, contingent, special, or punitive damages, including lost profits arising from the use or trust in this document (even if Alibaba Cloud has been notified of the possibility of such a loss).

5. By law, all the contents in Alibaba Cloud documents, including but not limited to pictures, architecture design, page layout, and text description, are intellectual property of Alibaba Cloud and/or its affiliates. This intellectual property includes, but is not limited to, trademark rights, patent rights, copyrights, and trade secrets. No part of this document shall be used, modified, reproduced, publicly transmitted, changed, disseminated, distributed, or published without the prior written consent of Alibaba Cloud and/or its affiliates. The names owned by Alibaba Cloud shall not be used, published, or reproduced for marketing, advertising, promotion, or other purposes without the prior written consent of Alibaba Cloud. The names owned by Alibaba Cloud include, but are not limited to, "Alibaba Cloud", "Aliyun", "HiChina", and other brands of Alibaba Cloud and/or its affiliates, which appear separately or in combination, as well as the auxiliary signs and patterns of the preceding brands, or anything similar to the company names, trade names, trademarks, product or service names, domain names, patterns, logos, marks, signs, or special descriptions that third parties identify as Alibaba Cloud and/or its affiliates.

6. Please directly contact Alibaba Cloud for any errors of this document.

Document conventions

Style	Description	Example
Danger	A danger notice indicates a situation that will cause major system changes, faults, physical injuries, and other adverse results.	Danger: Resetting will result in the loss of user configuration data.
Warning	A warning notice indicates a situation that may cause major system changes, faults, physical injuries, and other adverse results.	Warning: Restarting will cause business interruption. About 10 minutes are required to restart an instance.
Notice	A caution notice indicates warning information, supplementary instructions, and other content that the user must understand.	Notice: If the weight is set to 0, the server no longer receives new requests.
Note	A note indicates supplemental instructions, best practices, tips, and other content.	Note: You can use Ctrl + A to select all files.
>	Closing angle brackets are used to indicate a multi-level menu cascade.	Click **Settings**> **Network**> **Set network type**.
Bold	Bold formatting is used for buttons , menus, page names, and other UI elements.	Click **OK**.
`Courier font`	Courier font is used for commands	Run the `cd /d C:/window` command to enter the Windows system folder.
Italic	Italic formatting is used for parameters and variables.	`bae log list --instanceid` *Instance_ID*
[] or [a|b]	This format is used for an optional value, where only one item can be selected.	`ipconfig [-all|-t]`
{} or {a|b}	This format is used for a required value, where only one item can be selected.	`switch {active|stand}`

Table of Contents

1.What is ApsaraDB RDS? 06
2.Log on to the ApsaraDB RDS console 07
3.Quick Start 08
3.1. Procedure 08
3.2. Create an instance 08
3.3. Configure an IP address whitelist for an ApsaraDB RDS in... 11
3.4. Connect to an instance 12
3.5. Create an account 13
3.6. Create a database 15
4.Instances 16
4.1. Create an instance 16
4.2. View basic information of an instance 18
4.3. Restart an instance 18
4.4. Change the specifications of an instance 19
4.5. Set a maintenance window 19
4.6. Configure primary/secondary switchover 20
4.7. Release an instance 21
4.8. Read-only instances 21
4.8.1. Overview of read-only ApsaraDB RDS for SQL Server in... 21
4.8.2. Create a read-only ApsaraDB RDS for SQL Server insta... 22
4.8.3. View details of read-only instances 24
5.Accounts 25
5.1. Create an account 25
5.2. Reset the password 26
6.Databases 28
6.1. Create a database 28

6.2. Delete a database 28
6.3. Change the character set collation and the time zone of s... 30
6.4. Replicate databases between ApsaraDB RDS instances 33
7.Database connection 36
7.1. Change a vSwitch 36
7.2. Change the endpoint and port number of an instance 36
7.3. Apply for or release a public endpoint 37
7.4. Connect to an instance 38
8.Monitoring and alerting 40
8.1. View resource and engine monitoring data 40
9.Data security 42
9.1. Configure an IP address whitelist for an ApsaraDB RDS ins... 42
9.2. Configure SSL encryption 43
9.3. Configure TDE 46
10.Service availability 48
10.1. Switch workloads between primary and secondary Apsara... 48
11.Database backup and restoration 50
11.1. Configure an automatic backup policy 50
11.2. Manually back up an instance 50
11.3. Shrink transaction logs 51
12.Migrate full backup data to ApsaraDB RDS for SQL Server 52

1.What is ApsaraDB RDS?

ApsaraDB RDS is a stable, reliable, and scalable online database service. Based on the high-performance storage, ApsaraDB RDS provides a set of solutions for disaster recovery, backup, restoration, monitoring, and migration.

ApsaraDB RDS supports four database engines: MySQL, SQL Server, PolarDB, and PostgreSQL. You can create database instances based on these database engines to meet your business requirements. This topic describes the SQL Server engine.

ApsaraDB RDS for SQL Server

ApsaraDB RDS for SQL Server provides strong support for a variety of enterprise applications under the high-availability architecture. ApsaraDB RDS for SQL Server can also restore data to a specific point in time, which reduces costs.

ApsaraDB RDS for SQL Server provides basic features such as whitelist configuration, backup and restoration, transparent data encryption, data migration, and management for instances, accounts, and databases.

2.Log on to the ApsaraDB RDS console

This topic describes how to log on to the ApsaraDB RDS console.

Prerequisites

- Before you log on to the Apsara Uni-manager Management Console, the endpoint of the console is obtained from the deployment staff.
- We recommend that you use Google Chrome.

Procedure

1. Enter your username and password.

   Obtain the username and password that you use to log on to the Apsara Uni-manager Management Console from the operations administrator.

   > **Note** The first time that you log on to the Apsara Uni-manager Management Console, you must change the password of your account. For security purposes, your password must meet the minimum complexity requirements. The password must be 10 to 32 characters in length and must contain at least two of the following character types:
   >
   > - Uppercase or lowercase letters
   > - Digits
   > - Special characters, including exclamation points (!), at signs (@), number signs (#), dollar signs ($), and percent signs (%)

2. Click **Log On**.
3. If your account has multi-factor authentication (MFA) enabled, perform corresponding operations in the following scenarios:
   - You log on to the Apsara Uni-manager Management Console for the first time after MFA is enabled by the administrator:
     a. On the Bind Virtual MFA Device page, bind an MFA device.
     b. Enter the username and password again as in Step 2 and click **Log On**.
     c. Enter a six-digit MFA verification code and click **Authenticate**.
   - You have enabled MFA and bound an MFA device:

     Enter a six-digit MFA verification code and click **Authenticate**.

     > **Note** For more information, see the *Bind a virtual MFA device to enable MFA* topic in *Apsara Uni-manager Management Console User Guide*.
4. In the top navigation bar, choose **Products > Database Services > ApsaraDB RDS**.

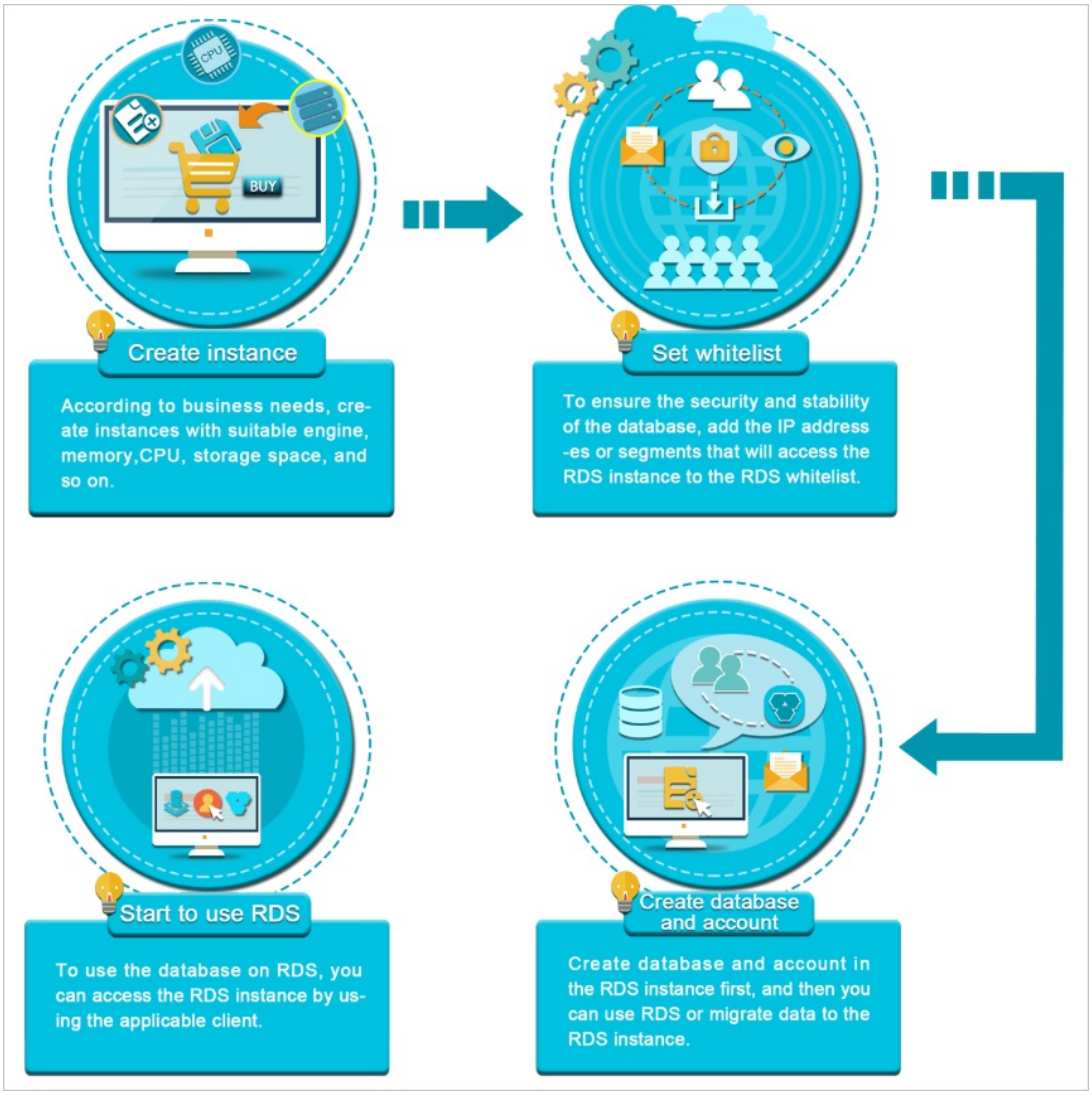
3.Quick Start

3.1. Procedure

ApsaraDB RDS quick start covers the following topics: creating an ApsaraDB RDS instance, configuring an IP address whitelist, creating a database, creating an account, and connecting to the instance.

The following figure shows the operations that you must perform before you use an ApsaraDB RDS instance.

Quick start flowchart



3.2. Create an instance

This topic describes how to create an instance in the ApsaraDB RDS console.

Prerequisites

An Apsara Stack tenant account is created.

Procedure

1. Log on to the ApsaraDB for RDS console.
2. On the **Instances** page, click **Create Instance** in the upper-right corner.
3. Configure the parameters described in the following table.

Section	Parameter	Description
Basic Configurations	**Organization**	The organization to which the instance belongs.
	Resource Set	The resource set to which the instance belongs.
Area	**Region**	The region in which you want to create the instance. Services in different regions cannot communicate over an internal network. After the instance is created, the region cannot be changed.
	Primary Node Zone	The zone in which the primary instance is deployed.
	Deployment Method	Specifies whether to deploy the primary and secondary instances in separate zones. ApsaraDB RDS supports **Multi-zone Deployment** and **Single-zone Deployment**. If you select **Multi-zone Deployment**, you must configure **Secondary Node Zone**.
Specifications	**Database Engine**	The database engine of the instance. Select **SQLServer**.
	Engine Version	The version of the database engine. Valid values: ○ **2012_ent_ha**: SQL Server 2012 EE ○ **2012_std_ha**: SQL Server 2012 SE ○ **2016_ent_ha**: SQL Server 2016 EE ○ **2016_std_ha**: SQL Server 2016 SE ○ **2017_ent_ha**: SQL Server 2017 EE
	Edition	The edition of the instance. For more information, see Instance types in *ApsaraDB RDS Product Introduction*.
	Instance Specifications	The specifications of the instance. The maximum number of connections and the maximum IOPS vary based on the memory size. The actual specifications are displayed in the console. For more information, see Instance types in *ApsaraDB RDS Product Introduction*.
	Storage Capacity	The storage capacity that is provided to store data files, system files, binlog files, and transaction files in the instance. For more information, see Instance types in *ApsaraDB RDS Product Introduction*.

Section	Parameter	Description
Network Type	Connection Type	The connection type of the instance. Valid values: ○ **Internet**: Instances of this connection type can be connected over the Internet. ○ **Internal Network**: Instances of this connection type can be connected over an internal network. **Note** After the instance is created, the value of this parameter cannot be changed. Proceed with caution.
	Network Type	The network type of the instance. Valid values: ○ **Classic Network**: Cloud services in the classic network are not isolated. Unauthorized access to a cloud service is blocked only by the security group or whitelist policy of the service. ○ **VPC**: A virtual private cloud (VPC) helps you build an isolated network environment on Apsara Stack. You can customize route tables, CIDR blocks, and gateways in a VPC. We recommend that you select VPC for improved security.
	VPC	The VPC in which you want to create the instance. **Note** When **Network Type** is set to **VPC**, you must specify this parameter.
	VSwitch	The vSwitch to which the instance is connected. **Note** When **Network Type** is set to **VPC**, you must specify this parameter.
	IP Whitelist	The IP address or CIDR block that is allowed to connect to the instance.
Instance Settings	Quantity	The number of instances that you want to create. Default value: 1.
	Instance Name	The name of the instance. ○ The name must be 2 to 64 characters in length. ○ The name must start with a letter. ○ The name can contain letters, digits, and the following special characters: _ - : ○ The name cannot start with http:// or https://.

4. Click **Submit**.

3.3. Configure an IP address whitelist for an ApsaraDB RDS instance

After you create an ApsaraDB RDS instance, you must add the IP addresses or CIDR blocks that are used for database access to the IP address whitelist of the instance to ensure database security and reliability.

Context

IP address whitelists make your ApsaraDB RDS instance more secure and do not interrupt the operations of your ApsaraDB RDS instance when you configure whitelists. We recommend that you maintain your IP address whitelists on a regular basis.

Procedure

1. Log on to the ApsaraDB for RDS console.
2. On the **Instances** page, find the target instance.
3. Click the instance ID or click **Manage** in the Actions column corresponding to the instance to go to the **Basic Information** page.
4. In the left-side navigation pane, click **Data Security**.
5. On the **Whitelist Settings** tab, click **Edit** corresponding to the **default** whitelist.

   > **Note** If you want to connect an ECS instance to an ApsaraDB RDS instance by using an internal endpoint, you must make sure that the two instances are in the same region and have the same network type. Otherwise, the connection fails.

6. In the **Edit Whitelist** dialog box, enter the IP addresses or CIDR blocks that are allowed to access the instance and click **OK**.

   > **Note**
   > - Limits for IP address whitelists:
   >   - You must separate multiple IP addresses with commas (,). A maximum of 1,000 different IP addresses can be added.
   >
   >   Supported formats are `0.0.0.0/0`, IP addresses such as `10.23.12.24`, or CIDR blocks such as `10.23.12.24/24`. `/24` indicates the length of the IP address prefix in the CIDR block. An IP address prefix can contain 1 to 32 bits.
   >   - If an IP address whitelist is empty or contains only `0.0.0.0/0`, all IP addresses can access the ApsaraDB RDS instance. This may cause security risks to the instance. Proceed with caution.
   > - If you click **Add Internal IP Addresses of ECS Instances**, the IP addresses of all the Elastic Compute Service (ECS) instances that are created within your Apsara Stack tenant account appear. Then, you can select the IP addresses and add them to an IP address whitelist.

3.4. Connect to an instance

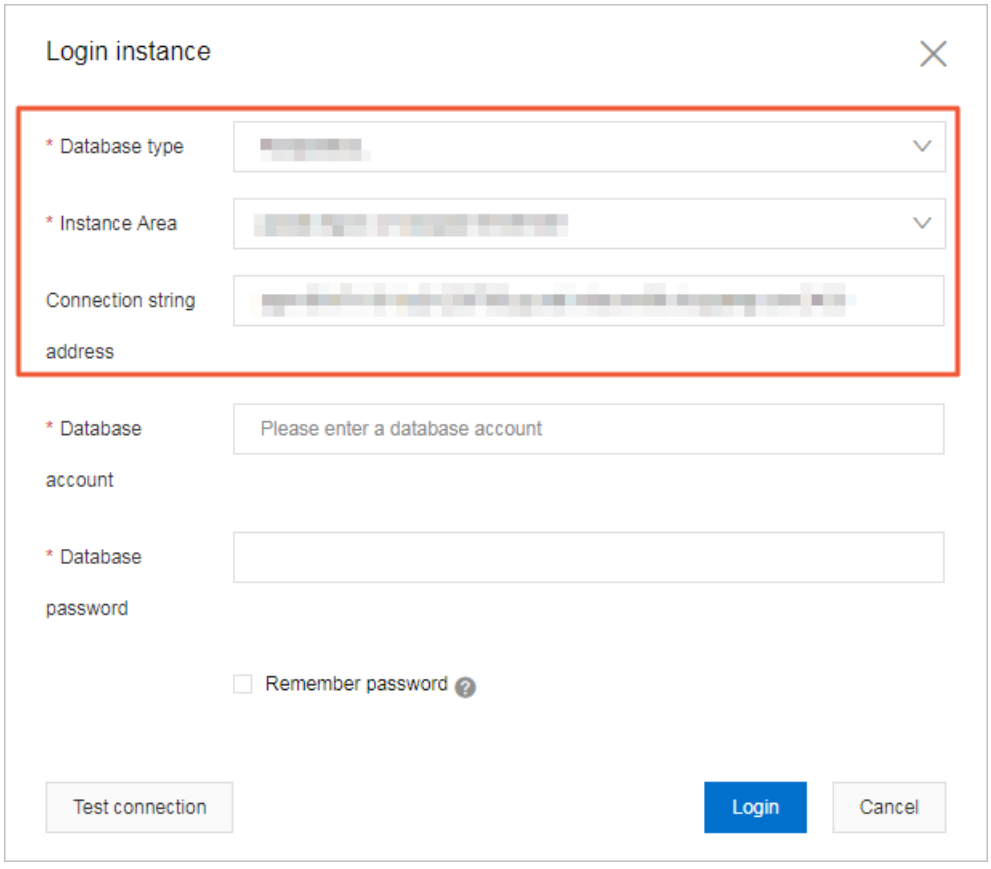
This topic describes how to use Data Management (DMS) to connect to an ApsaraDB RDS instance.

Prerequisites

- A database is created. For more information, see Create a database.
- A database account is created. For more information, see Create an account.

Procedure

1. Log on to the ApsaraDB for RDS console.
2. On the **Instances** page, find the target instance.
3. Click the instance ID or click **Manage** in the Actions column corresponding to the instance to go to the **Basic Information** page.
4. Click **Log On to DB** in the upper-right corner of the page.
5. In the **Login instance** dialog box of the **DMS** console, check values of **Database type**, **Instance Area**, and **Connection string address**. If the information is correct, enter **Database account** and **Database password**, as shown in the following figure.



Parameter	Description
Database type	The engine of the database. By default, the engine of the database to be connected is displayed.

Parameter	Description
Instance Area	The region where the instance is deployed. By default, the region of the current instance is displayed.
Connection string address	The endpoint of the instance. By default, the endpoint of the current instance is displayed.
Database account	The account of the database to be connected.
Database password	The password of the account used to connect to the database.

6. Click **Login**.

> Note
> - If you want the browser to remember the password, select **Remember password** and click **Login**.
> - If you cannot connect to the instance, check the IP address whitelist settings. For more information, see Configure a whitelist.

3.5. Create an account

This topic describes how to create an account on an ApsaraDB RDS for SQL Server instance.

Prerequisites

The instance is in the **Running** state.

Procedure

1. Log on to the ApsaraDB for RDS console.
2. On the **Instances** page, find the target instance.
3. Click the instance ID or click **Manage** in the Actions column corresponding to the instance to go to the **Basic Information** page.
4. In the left-side navigation pane, click **Accounts**.
5. On the right side of the page, click **Create Account**.
6. Enter the information of the account that you want to create.

Parameter	Description
Database Account	Enter the name of the account. The name must be 2 to 16 characters in length and can contain lowercase letters, digits, and underscores (_). It must start with a lowercase letter and end with a lowercase letter or a digit.

Parameter	Description
Account Type	○ **Privileged Account**: You can select the **Privileged Account** option only if you create an account on your ApsaraDB RDS instance for the first time. Each ApsaraDB RDS instance can have only a single privileged account. The privileged account of an ApsaraDB RDS instance cannot be deleted. ○ **Standard Account**: You can select the **Standard Account** option only after a privileged account is created on your ApsaraDB RDS instance. Each ApsaraDB RDS instance can have more than one standard account. You must manually grant the permissions on databases to each standard account.
Authorized Databases (available only for standard accounts)	Select the authorized databases of the account when the **Standard Account** type is selected. If no databases are created, you can leave this parameter empty. You can perform the following steps to grant permissions on more than one database to the account: i. In the **Unauthorized Databases** section, select the databases on which you want to grant permissions to the account. ii. Click **Add** to add the selected databases to the **Authorized Databases** section. iii. In the Authorized Databases section, specify the permissions that the account is granted on each authorized database. The permissions can be **Read/Write**, **Read-only**, or **Owner**. You can also click **Set All to Read/Write**, **Set All to Read-only**, or **Set All to Owner** to set the permissions of the account on all authorized databases. Note ■ The account is authorized to create tables, delete tables, and modify schemas in a database only when it has the **Owner** permission on the database. ■ The account has permissions on all databases and does not require authorization if you select the **Privileged Account** type.
Password	Enter the password of the account. The password must meet the following requirements: ○ The password is 8 to 32 characters in length. ○ The password contains at least three of the following character types: uppercase letters, lowercase letters, digits, and special characters. ○ Special characters include `! @ # $ % ^ & * ( ) _ + - =`

Parameter	Description
Re-enter Password	Enter the password of the account again.
Description	Enter a description that helps identify the account. The description can be up to 256 characters in length.

7. Click **Create**.

3.6. Create a database

This topic describes how to create a database on an ApsaraDB RDS for SQL Server instance in the ApsaraDB RDS console.

Procedure

1. Log on to the ApsaraDB for RDS console.
2. On the **Instances** page, find the target instance.
3. Click the instance ID or click **Manage** in the Actions column corresponding to the instance to go to the **Basic Information** page.
4. In the left-side navigation pane, click **Databases**.
5. In the upper-right corner of the page, click **Create Database**.
6. Configure the parameters for the database that you want to create.

Parameter	Description
Database Name	Enter the name of the database. The name must be 2 to 64 characters in length. It can contain lowercase letters, digits, underscores (_), and hyphens (-). It must start with a lowercase letter and end with a lowercase letter or digit.
Supported Character Sets	Select the character set that is supported by the database. You can also select **all** and then select a character set from the drop-down list that appears.
Description	Enter a description of the database to facilitate subsequent management. The description can be up to 256 characters in length.

7. Click **Create**.

4.Instances

4.1. Create an instance

This topic describes how to create an instance in the ApsaraDB RDS console.

Prerequisites

An Apsara Stack tenant account is created.

Procedure

1. Log on to the ApsaraDB for RDS console.
2. On the **Instances** page, click **Create Instance** in the upper-right corner.
3. Configure the parameters described in the following table.

Section	Parameter	Description
Basic Configurations	**Organization**	The organization to which the instance belongs.
	Resource Set	The resource set to which the instance belongs.
Area	**Region**	The region in which you want to create the instance. Services in different regions cannot communicate over an internal network. After the instance is created, the region cannot be changed.
	Primary Node Zone	The zone in which the primary instance is deployed.
	Deployment Method	Specifies whether to deploy the primary and secondary instances in separate zones. ApsaraDB RDS supports **Multi-zone Deployment** and **Single-zone Deployment**. If you select **Multi-zone Deployment**, you must configure **Secondary Node Zone**.
Specifications	**Database Engine**	The database engine of the instance. Select **SQLServer**.
	Engine Version	The version of the database engine. Valid values: ○ **2012_ent_ha**: SQL Server 2012 EE ○ **2012_std_ha**: SQL Server 2012 SE ○ **2016_ent_ha**: SQL Server 2016 EE ○ **2016_std_ha**: SQL Server 2016 SE ○ **2017_ent_ha**: SQL Server 2017 EE
	Edition	The edition of the instance. For more information, see Instance types in *ApsaraDB RDS Product Introduction*.

Section	Parameter	Description
	Instance Specifications	The specifications of the instance. The maximum number of connections and the maximum IOPS vary based on the memory size. The actual specifications are displayed in the console. For more information, see Instance types in *ApsaraDB RDS Product Introduction*.
	Storage Capacity	The storage capacity that is provided to store data files, system files, binlog files, and transaction files in the instance. For more information, see Instance types in *ApsaraDB RDS Product Introduction*.
Network Type	**Connection Type**	The connection type of the instance. Valid values: ○ **Internet**: Instances of this connection type can be connected over the Internet. ○ **Internal Network**: Instances of this connection type can be connected over an internal network. **Note** After the instance is created, the value of this parameter cannot be changed. Proceed with caution.
	Network Type	The network type of the instance. Valid values: ○ **Classic Network**: Cloud services in the classic network are not isolated. Unauthorized access to a cloud service is blocked only by the security group or whitelist policy of the service. ○ **VPC**: A virtual private cloud (VPC) helps you build an isolated network environment on Apsara Stack. You can customize route tables, CIDR blocks, and gateways in a VPC. We recommend that you select VPC for improved security.
	VPC	The VPC in which you want to create the instance. **Note** When **Network Type** is set to **VPC**, you must specify this parameter.
	VSwitch	The vSwitch to which the instance is connected. **Note** When **Network Type** is set to **VPC**, you must specify this parameter.
	IP Whitelist	The IP address or CIDR block that is allowed to connect to the instance.
	Quantity	The number of instances that you want to create. Default value: 1.

Section	Parameter	Description
Instance Settings	Instance Name	The name of the instance. ○ The name must be 2 to 64 characters in length. ○ The name must start with a letter. ○ The name can contain letters, digits, and the following special characters: _ - : ○ The name cannot start with http:// or https://.

4. Click **Submit**.

4.2. View basic information of an instance

This topic describes how to view the details of an ApsaraDB RDS instance, such as its basic information, internal network connection information, status, and configurations.

Procedure

1. Log on to the ApsaraDB for RDS console.
2. Use one of the following methods to go to the **Basic Information** page of an instance:
   - On the **Instances** page, click the ID of an instance to go to the **Basic Information** page.
   - On the **Instances** page, click **Manage** in the **Actions** column corresponding to an instance to go to the **Basic Information** page.

4.3. Restart an instance

This topic describes how to manually restart an ApsaraDB RDS instance. This applies if the number of connections exceeds the specified threshold or if an instance has performance issues.

Prerequisites

The instance is in the **Running** state.

Procedure

1. Log on to the ApsaraDB for RDS console.
2. On the **Instances** page, find the instance that you want to manage.
3. Click **Restart Instance** in the upper-right corner.

   > **Note** When you restart an instance, applications are disconnected from the instance. We recommend that you make appropriate service arrangements before you restart an instance. Proceed with caution.

4. In the Restart Instance message, click **Confirm**.

4.4. Change the specifications of an instance

This topic describes how to change specifications such as the instance type and storage space if they do not meet the requirements of your application. When the specification changes take effect, a 30-second network interruption may occur. Business operations that involve databases, accounts, and networks are interrupted. We recommend that you change the specifications during off-peak hours or make sure that your applications are configured with automatic reconnection policies.

Procedure

1. Log on to the ApsaraDB for RDS console.
2. On the **Instances** page, find the target instance.
3. Click the instance ID or click **Manage** in the Actions column corresponding to the instance to go to the **Basic Information** page.
4. In the **Configuration Information** section, click **Change Specifications**.
5. On the **Change Specifications** page, specify **Instance Type** and **Storage Capacity**.
6. After you configure the preceding parameters, click **Submit**.

4.5. Set a maintenance window

This topic describes how to set the maintenance window of an ApsaraDB RDS for SQL Server instance. The backend system performs maintenance on the ApsaraDB RDS instance during the maintenance window. This ensures the stability of the ApsaraDB RDS instance. The default maintenance window is from 02:00 (UTC+8) to 06:00 (UTC+8). We recommend that you set the maintenance window to off-peak hours of your business to avoid impacts on your business.

Context

- An instance enters the **Maintaining Instance** state before the maintenance window to ensure stability during the maintenance process. When the instance is in this state, access to data in the database and query operations such as performance monitoring are not affected. However, except for account and database management and IP address whitelist configuration, modification operations such as upgrade, downgrade, and restart are temporarily unavailable.
- During the maintenance window, one or two network interruptions may occur. Make sure that your applications are configured with automatic reconnection policies.

Procedure

1. Log on to the ApsaraDB for RDS console.
2. On the **Instances** page, find the target instance.
3. Click the instance ID or click **Manage** in the Actions column corresponding to the instance to go to the **Basic Information** page.
4. In the **Configuration Information** section, click **Configure** to the right of **Maintenance Window**.
5. Select a maintenance window and click **Save**.

> **Note** The maintenance window is displayed in UTC+8.

4.6. Configure primary/secondary switchover

ApsaraDB RDS provides the primary/secondary switchover feature to ensure the high availability of databases. The primary/secondary switchover is performed when the primary instance becomes unavailable. You can also manually switch your business to the secondary instance.

Prerequisites

The instance is in the **Running** state.

Context

An ApsaraDB RDS for SQL Server instance has a secondary instance. Data is synchronized in real time between the primary and secondary instances. You can access only the primary instance. The secondary instance serves only as a backup and does not allow external access. If the primary instance cannot be accessed, your workloads are automatically switched over to the secondary instance. After the switchover, the primary instance becomes the secondary instance.

Notice

- You may encounter a network interruption during a switchover. Make sure that your application is configured to automatically reconnect to the instance.
- During a switchover, a 1-minute data quality protection mechanism is enabled for data synchronization. If the primary and secondary database states are incorrect or if the latency for data synchronization exceeds 1 minute due to SQL Server errors, the HA system does not automatically perform the primary/secondary switchover. You must determine whether to perform the switchover.
- If an instance is intermittently unavailable due to excessive mirroring event waits, the switchover is not performed. The instance automatically becomes available again.

Procedure

1. Log on to the ApsaraDB for RDS console.
2. On the **Instances** page, find the target instance.
3. Click the instance ID or click **Manage** in the Actions column corresponding to the instance to go to the **Basic Information** page.
4. In the left-side navigation pane, click **Service Availability**.
5. In the **Availability Information** section, click **Switch Primary/Secondary Instance**.
6. In the dialog box that appears, click **OK**.

Result

After the switchover is complete, the original primary instance becomes the secondary instance for the next primary/secondary switchover.

4.7. Release an instance

This topic describes how to manually release an instance.

Context

- Only instances in the running state can be manually released.
- After an instance is released, the instance data is immediately deleted. We recommend that you back up your data before you release an instance.

Procedure

1. Log on to the ApsaraDB for RDS console.
2. Find the instance that you want to release and choose **More > Release Instance**.
3. In the **Release Instance** message, click **Confirm**.

4.8. Read-only instances

4.8.1. Overview of read-only ApsaraDB RDS for SQL Server instances

This topic provides an overview of read-only ApsaraDB RDS for SQL Server instances. If a large number of read requests overwhelm the primary instance, your business may be interrupted. In this case, you can create one or more read-only instances to offload read requests from the primary instance. This scales the read capability of your database system and increases the throughput of your application.

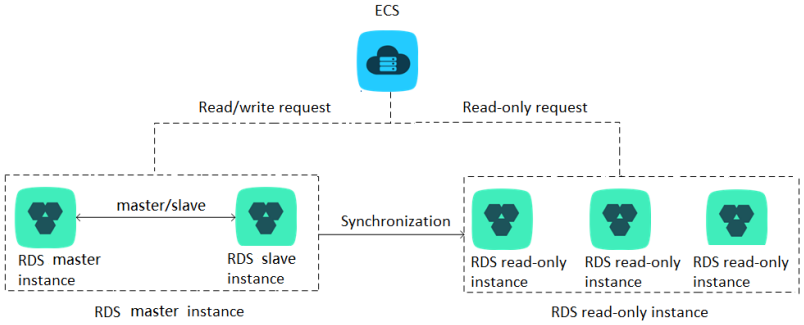
Overview

When a read-only instance is created, the data is replicated from the secondary instance. The data is consistent with that of the primary instance. Data updates of the primary instance are synchronized to all read-only instances.

> **Note**

- Only ApsaraDB RDS instances that run SQL Server 2017 EE support read-only instances.
- Each read-only instance works in a single-node architecture, where no instances are provided as backups.

The following figure shows the topology of read-only instances.



Features

- The specifications of a read-only instance can differ from the specifications of the primary instance, and can be changed at any time. We recommend that you select specifications of a read-only instance that are higher than or equal to those of the primary instance. If the specifications of a read-only instance are lower than those of the primary instance, the read-only instance may have high latency or workloads.
- Read-only instances do not require database or account maintenance, because their database and account information is synchronized with the primary instance.
- A read-only instance automatically replicates the IP address whitelists of the primary instance. However, the IP address whitelists for the read-only instance are independent of those of the primary instance. For information about how to modify the whitelists of a read-only instance, see Configure a whitelist.
- You can monitor up to 20 system performance metrics, such as the disk capacity, input/output operations per second (IOPS), number of connections, CPU utilization, and network traffic.

Limits

- You can create up to seven read-only instances.
- You cannot configure backup policies or manually create backups for read-only instances, because these are already configured or created on the primary instance.
- You cannot create a temporary instance by using a backup set or from a point in time. In addition, you cannot overwrite a read-only instance by using a backup set.
- After a read-only instance is created, you cannot use a data backup file to restore it in overwrite mode.
- You cannot migrate data to read-only RDS instances.
- You cannot create or delete databases on read-only instances.
- You cannot create or delete accounts, authorize accounts, or change the passwords of accounts on read-only instances.

FAQ

Can I manage the accounts created on the primary instance from its read-only instances?

No, although accounts created on the primary instance are replicated to its read-only instances, you cannot manage the accounts on the read-only instances. The accounts have only read permissions on the read-only instances.

4.8.2. Create a read-only ApsaraDB RDS for SQL Server instance

This topic describes how to create a read-only instance for your primary ApsaraDB RDS for SQL server instance. This allows your database system to process a large number of read requests and increases the throughput of your application. Each read-only ApsaraDB RDS instance is a replica of the primary instance. Data updates on the primary instance are synchronized to all the read-only instances.

Prerequisites

The primary instance runs SQL Server 2017 EE.

Precautions

- You can create read-only instances for the primary ApsaraDB RDS instance. However, you cannot convert existing ApsaraDB RDS instances into read-only instances.
- While you create a read-only instance, the system replicates data from a secondary instance. Therefore, the operation of your primary instance is not interrupted.
- You can create up to seven read-only instances.
- For more information about read-only ApsaraDB RDS instances, see Overview of read-only ApsaraDB RDS for SQL Server instances.

Create a read-only instance

1. Log on to the ApsaraDB for RDS console.
2. On the **Instances** page, find the target instance.
3. Click the instance ID or click **Manage** in the Actions column corresponding to the instance to go to the **Basic Information** page.
4. In the **Distributed by Instance Role** section on the right side of the page, click **Create Read-only Instance**.
5. Configure the following parameters and click **Submit**.

Section	Parameter	Description
Region	**Region**	The region in which you want to create the instance.
Specifications	**Database Engine**	The database engine of the read-only instance, which is the same as that of the primary instance and cannot be changed.
	Engine Version	The engine version of the read-only instance, which is the same as that of the primary instance and cannot be changed.
	Edition	Set the value to **Read-only**.
	Instance Type	The instance type of the read-only instance. The instance type of the read-only instance can be different from that of the primary instance, and can be changed at any time to facilitate flexible upgrade and downgrade. For more information, see Instance types in *ApsaraDB RDS Product Information*. **Note** To ensure sufficient I/O throughput for data synchronization, we recommend that you select at least the same instance type as the primary instance for read-only instances.

Section	Parameter	Description
	Storage Capacity	The storage space of the read-only instance. To ensure sufficient I/O throughput for data synchronization, we recommend that you select at least the same instance type and storage space as the primary instance for the read-only instance. For more information, see Instance types in *ApsaraDB RDS Product Introduction*.
Network Type	**Network Type**	The network type of the read-only instance, which is the same as that of the primary instance and cannot be changed.
	VPC	Select a VPC if the network type is set to VPC.
	vSwitch	Select a vSwitch if the network type is set to VPC.

4.8.3. View details of read-only instances

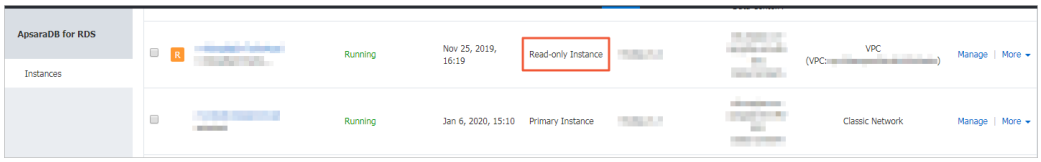
This topic describes how to view details of read-only instances. You can go to the Basic Information page of a read-only instance from the Instances page or the read-only instance list of the primary instance. Read-only instances are managed in the same way as primary instances. The read-only instance management page shows the management operations that can be performed.

View instance details by using a read-only instance

1. Log on to the ApsaraDB for RDS console.
2. On the **Instances** page, click the ID of a read-only instance. The **Basic Information** page appears.

   In the instance list, Instance Role of read-only instances is displayed as Read-only Instance, as shown in View a read-only instance.

   View a read-only instance



View instance details by using the primary instance

1. Log on to the ApsaraDB for RDS console.
2. On the **Instances** page, find the target instance.
3. Click the instance ID or click **Manage** in the Actions column corresponding to the instance to go to the **Basic Information** page.
4. On the **Basic Information** page, move the pointer over the number below **Read-only Instance** in the **Distributed by Instance Role** section. The ID of the read-only instance is displayed.
5. Click the ID of the read-only instance to go to the read-only instance management page.

5.Accounts

5.1. Create an account

This topic describes how to create an account on an ApsaraDB RDS for SQL Server instance.

Prerequisites

The instance is in the **Running** state.

Procedure

1. Log on to the ApsaraDB for RDS console.
2. On the **Instances** page, find the target instance.
3. Click the instance ID or click **Manage** in the Actions column corresponding to the instance to go to the **Basic Information** page.
4. In the left-side navigation pane, click **Accounts**.
5. On the right side of the page, click **Create Account**.
6. Enter the information of the account that you want to create.

Parameter	Description
Database Account	Enter the name of the account. The name must be 2 to 16 characters in length and can contain lowercase letters, digits, and underscores (_). It must start with a lowercase letter and end with a lowercase letter or a digit.
Account Type	◦ **Privileged Account**: You can select the **Privileged Account** option only if you create an account on your ApsaraDB RDS instance for the first time. Each ApsaraDB RDS instance can have only a single privileged account. The privileged account of an ApsaraDB RDS instance cannot be deleted. ◦ **Standard Account**: You can select the **Standard Account** option only after a privileged account is created on your ApsaraDB RDS instance. Each ApsaraDB RDS instance can have more than one standard account. You must manually grant the permissions on databases to each standard account.

Parameter	Description
Authorized Databases (available only for standard accounts)	Select the authorized databases of the account when the **Standard Account** type is selected. If no databases are created, you can leave this parameter empty. You can perform the following steps to grant permissions on more than one database to the account: i. In the **Unauthorized Databases** section, select the databases on which you want to grant permissions to the account. ii. Click **Add** to add the selected databases to the **Authorized Databases** section. iii. In the Authorized Databases section, specify the permissions that the account is granted on each authorized database. The permissions can be **Read/Write**, **Read-only**, or **Owner**. You can also click **Set All to Read/Write**, **Set All to Read-only**, or **Set All to Owner** to set the permissions of the account on all authorized databases. **Note** ■ The account is authorized to create tables, delete tables, and modify schemas in a database only when it has the **Owner** permission on the database. ■ The account has permissions on all databases and does not require authorization if you select the **Privileged Account** type.
Password	Enter the password of the account. The password must meet the following requirements: ○ The password is 8 to 32 characters in length. ○ The password contains at least three of the following character types: uppercase letters, lowercase letters, digits, and special characters. ○ Special characters include `! @ # $ % ^ & * ( ) _ + - =`
Re-enter Password	Enter the password of the account again.
Description	Enter a description that helps identify the account. The description can be up to 256 characters in length.

7. Click **Create**.

5.2. Reset the password

You can use the ApsaraDB RDS console to reset the password of your database account.

Prerequisites

The instance is in the **Running** state.

Procedure

1. Log on to the ApsaraDB for RDS console.

2. On the **Instances** page, find the instance that you want to manage.
3. In the left-side navigation pane, click **Accounts**.
4. Find an account and click **Reset Password** in the **Actions** column.
5. In the dialog box that appears, enter and confirm the new password, and then click **OK**.

> **Note** The password must meet the following requirements:
>
> - The password is 8 to 32 characters in length.
> - The password contains at least three of the following character types: uppercase letters, lowercase letters, digits, and special characters.
> - Special characters include
>
> ! @ # $ % ^ & * ( ) _ + - =

6.Databases

6.1. Create a database

This topic describes how to create a database on an ApsaraDB RDS for SQL Server instance.

Terms

- Instance: a virtualized database server on which you can create and manage more than one database.
- Database: a set of data that is stored in an organized manner and can be shared by a number of users. A database provides the minimal redundancy and is independent of applications. In simple words, a database is a data warehouse that is used to store data.
- Character set: a collection of letters, special characters, and encoding rules that are used in a database.

Prerequisites

An ApsaraDB RDS for SQL Server instance is created. For more information, see Create an instance.

Procedure

For more information, see Create a database.

6.2. Delete a database

This topic describes how to delete a database from an ApsaraDB RDS for SQL Server instance. You can delete a database by using the ApsaraDB RDS console or an SQL statement.

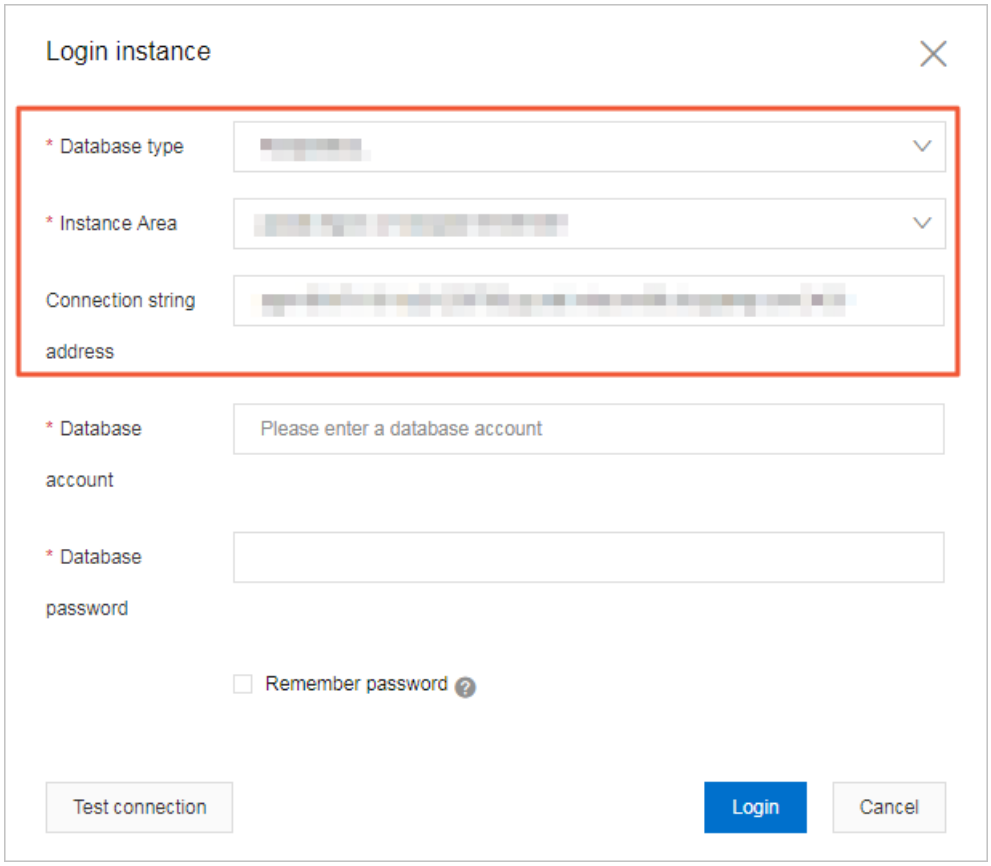
Use the console to delete a database

1. Log on to the ApsaraDB for RDS console.
2. On the **Instances** page, find the target instance.
3. Click the instance ID or click **Manage** in the Actions column corresponding to the instance to go to the **Basic Information** page.
4. In the left-side navigation pane, click **Databases**.
5. Find the database that you want to delete and click **Delete** in the **Actions** column.
6. In the message that appears, click **Confirm**.

Execute an SQL statement to delete a database

1. Log on to the ApsaraDB for RDS console.
2. On the **Instances** page, find the target instance.
3. Click the instance ID or click **Manage** in the Actions column corresponding to the instance to go to the **Basic Information** page.
4. Click **Log On to DB** in the upper-right corner of the page.
5. In the **Login instance** dialog box of the **DMS** console, check values of **Database type**, **Instance Area**, and **Connection string address**. If the information is correct, enter **Database account**

and **Database password**, as shown in the following figure.



Parameter	Description
Database type	The engine of the database. By default, the engine of the database to be connected is displayed.
Instance Area	The region where the instance is deployed. By default, the region of the current instance is displayed.
Connection string address	The endpoint of the instance. By default, the endpoint of the current instance is displayed.
Database account	The account of the database to be connected.
Database password	The password of the account used to connect to the database.

6. Click **Login**.

> **Note**
>
> - If you want the browser to remember the password, select **Remember password** and click **Login**.
> - If you cannot connect to the instance, check the IP address whitelist settings. For more information, see Configure a whitelist.

7. The **SQLConsole** page appears after you log on to the instance. Execute a statement in the

following format to delete a database:

```
drop database <database name>;
```

> **Note** If the instance runs SQL Server 2012 or later on RDS High-availability Edition, you can also use the following stored procedure. This stored procedure deletes the specified database, removes the associated image, and closes the connection to the database.
>
> ```
> EXEC sp_rds_drop_database 'database name'
> ```

8. Click **execute**.

6.3. Change the character set collation and the time zone of system databases

This topic describes how to change the character set collation and the time zone of system databases. System databases include master, msdb, tempdb, and model.

Prerequisites

- The instance runs SQL Server 2012, 2016, or 2017.
- No database other than system databases exists on the instance.

> **Note** If you have just deleted databases from the instance, the deletion task may be pending in the secondary instance. Before you change the character set collation and the time zone, make sure that the primary and secondary instances do not contain databases.

Precautions

- The default character set collation is Chinese_PRC_CI_AS.
- The default time zone is China Standard Time.
- You can view the available character set collations and time zones in the console.
- The instance is in the unavailable state during the change process. It takes about 1 minute to change the time zone, and 2 to 10 minutes to change the character set collation.

Procedure

1. Log on to the ApsaraDB for RDS console.
2. On the **Instances** page, find the target instance.
3. Click the instance ID or click **Manage** in the Actions column corresponding to the instance to go to the **Basic Information** page.
4. In the left-side navigation pane, click **Databases**.
5. On the Databases page, click **Change Character Set Collation and Time Zone**.

> **Note** If you fail to find this button on the page, make sure that the requirements in Prerequisites are met.

6. In the dialog box that appears, select **Time Zone**, **Character Set Collation**, or both of them, and click **OK**.

UTC offsets of time zones

Time zone	UTC offset	Description
Afghanistan Standard Time	(UTC+04:30)	Kabul
Alaskan Standard Time	(UTC-09:00)	Alaska
Arabian Standard Time	(UTC+04:00)	Abu Dhabi, Muscat
Atlantic Standard Time	(UTC-04:00)	Atlantic Time (Canada)
AUS Central Standard Time	(UTC+09:30)	Darwin
AUS Eastern Standard Time	(UTC+10:00)	Canberra, Melbourne, Sydney
Belarus Standard Time	(UTC+03:00)	Minsk
Canada Central Standard Time	(UTC-06:00)	Saskatchewan
Cape Verde Standard Time	(UTC-01:00)	Cabo Verde Is.
Cen. Australia Standard Time	(UTC+09:30)	Adelaide
Central America Standard Time	(UTC-06:00)	Central America
Central Asia Standard Time	(UTC+06:00)	Astana
Central Brazilian Standard Time	(UTC-04:00)	Cuiaba
Central Europe Standard Time	(UTC+01:00)	Belgrade, Bratislava, Budapest, Ljubljana, Prague
Central European Standard Time	(UTC+01:00)	Sarajevo, Skopje, Warsaw, Zagreb
Central Pacific Standard Time	(UTC+11:00)	Solomon Islands, New Caledonia
Central Standard Time	(UTC-06:00)	Central Time (US and Canada)
Central Standard Time (Mexico)	(UTC-06:00)	Guadalajara, Mexico City, Monterrey
China Standard Time	(UTC+08:00)	Beijing, Chongqing, Hong Kong, Urumqi
E. Africa Standard Time	(UTC+03:00)	Nairobi

Time zone	UTC offset	Description
E. Australia Standard Time	(UTC+10:00)	Brisbane
E. Europe Standard Time	(UTC+02:00)	Chisinau
E. South America Standard Time	(UTC-03:00)	Brasilia
Eastern Standard Time	(UTC-05:00)	Eastern Time (US and Canada)
Georgian Standard Time	(UTC+04:00)	Tbilisi
GMT Standard Time	(UTC)	Dublin, Edinburgh, Lisbon, London
Greenland Standard Time	(UTC-03:00)	Greenland
Greenwich Standard Time	(UTC)	Monrovia, Reykjavik
GTB Standard Time	(UTC+02:00)	Athens, Bucharest
Hawaiian Standard Time	(UTC-10:00)	Hawaii
India Standard Time	(UTC+05:30)	Chennai, Kolkata, Mumbai, New Delhi
Jordan Standard Time	(UTC+02:00)	Amman
Korea Standard Time	(UTC+09:00)	Seoul
Middle East Standard Time	(UTC+02:00)	Beirut
Mountain Standard Time	(UTC-07:00)	Mountain Time (US and Canada)
Mountain Standard Time (Mexico)	(UTC-07:00)	Chihuahua, La Paz, Mazatlan
US Mountain Standard Time	(UTC-07:00)	Arizona
New Zealand Standard Time	(UTC+12:00)	Auckland, Wellington
Newfoundland Standard Time	(UTC-03:30)	Newfoundland
Pacific SA Standard Time	(UTC-03:00)	Santiago
Pacific Standard Time	(UTC-08:00)	Pacific Time (US and Canada)
Pacific Standard Time (Mexico)	(UTC-08:00)	Baja California
Russian Standard Time	(UTC+03:00)	Moscow, St. Petersburg, Volgograd
SA Pacific Standard Time	(UTC-05:00)	Bogota, Lima, Quito, Rio Branco
SE Asia Standard Time	(UTC+07:00)	Bangkok, Hanoi, Jakarta

Time zone	UTC offset	Description
China Standard Time	(UTC+08:00)	Kuala Lumpur, Singapore
Tokyo Standard Time	(UTC+09:00)	Osaka, Sapporo, Tokyo
US Eastern Standard Time	(UTC-05:00)	Indiana (East)
UTC	UTC	Coordinated Universal Time
UTC-02	(UTC-02:00)	Coordinated Universal Time-02
UTC-08	(UTC-08:00)	Coordinated Universal Time-08
UTC-09	(UTC-09:00)	Coordinated Universal Time-09
UTC-11	(UTC-11:00)	Coordinated Universal Time-11
UTC+12	(UTC+12:00)	Coordinated Universal Time+12
W. Australia Standard Time	(UTC+08:00)	Perth
W. Central Africa Standard Time	(UTC+01:00)	West Central Africa
W. Europe Standard Time	(UTC+01:00)	Amsterdam, Berlin, Bern, Rome, Stockholm, Vienna

6.4. Replicate databases between ApsaraDB RDS instances

This topic describes how to replicate databases between ApsaraDB RDS instances that run SQL Server 2012 or SQL Server 2016.

Prerequisites

The source and destination instances must meet the following requirements:

- The source and destination instances belong to the same Apsara Stack tenant account.
- The engine version of the destination instance is not earlier than that of the source instance.
- The source and destination instances reside in the same region and use the same network type. Their zones can be different.
- The destination instance does not have databases whose names are the same as those of the databases that you want to replicate.
- The available storage capacity of the destination instance is larger than the size of the databases that you want to replicate from the source instance.

Context

During the replication process, ApsaraDB RDS first performs a full backup of the source instance and then replicates the full data to the destination instance. If data is written to the source instance during the replication, incremental data of the source instance is not replicated to the destination instance.

You can choose to replicate a single database or all databases in the source instance. If the replication task fails, no data is transferred to the destination instance. This ensures data consistency.

Procedure

1. Log on to the ApsaraDB for RDS console.
2. On the **Instances** page, find the instance that you want to manage.
3. In the left-side navigation pane, click **Databases**.
4. Click **Replicate to Another Instance**. In the dialog box that appears, configure the parameters described in the following table.

Parameter	Description
Source Instance Name	The ID of the source instance.
Target Instance Name	The ID of the destination instance. The drop-down list displays all the instances that reside in the same region and use the same SQL Server version as the source instance. You can select a destination instance from the drop-down list. Note

Parameter	Description
Source Databases	The databases that you want to replicate to the destination instance. You can click the > or < icon to add or remove databases. **Note** You can select multiple databases to batch add or remove databases. Make sure that the following conditions are met: ○ The available storage capacity of the destination instance is larger than the size of the databases that you want to replicate from the source instance. ○ The destination instance does not have databases whose names are the same as those of the databases that you want to replicate. **Note** ○ During the replication process, database accounts and account permissions are also replicated from the source instance to the destination instance. ■ If the destination instance has accounts whose usernames are the same as those of accounts on the source instance, the accounts on the destination instance are granted the same permissions as the accounts on the source instance. ■ If the destination instance does not have the same accounts, the accounts are first created on the destination instance and then granted the same permissions as the accounts on the source instance. ○ If the source and destination instances have databases whose names are the same, these databases are not replicated.

5. Click **OK**.

7.Database connection

7.1. Change a vSwitch

This topic describes how to change a vSwitch for an ApsaraDB RDS instance that is deployed in a VPC.

Prerequisites

The instance is deployed in a VPC.

Procedure

1. Log on to the ApsaraDB for RDS console.
2. On the **Instances** page, find the instance that you want to manage.
3. In the left-side navigation pane, click **Database Connection**.
4. In the upper-right corner of the Database Connection section, click **Switch vSwitch**.
5. In the dialog box that appears, select a vSwitch and click **OK**.
6. In the message that appears, click **Switch**.

> **Note**
>
> - You may encounter a network interruption of about 30 seconds during the change process. Make sure that your application is configured to automatically reconnect to the instance.
> - We recommend that you clear the cache immediately after the instance is switched to a new VPC and vSwitch. Otherwise, data can only be read but cannot be written.

7.2. Change the endpoint and port number of an instance

This topic describes how to view and change the endpoint and port number of an ApsaraDB RDS instance.

View the endpoint and port number

1. Log on to the ApsaraDB for RDS console.
2. On the **Instances** page, find the instance that you want to manage.
3. In the **Basic Information** section, view the internal and public endpoints and port numbers.

Change the endpoint and port number of an instance

1. Log on to the ApsaraDB for RDS console.
2. On the **Instances** page, find the instance that you want to manage.
3. In the left-side navigation pane, click **Database Connection**.
4. In the upper-right corner of the Database Connection section, click **Change Endpoint**.

5. In the dialog box that appears, set Connection Type, Endpoint, and Port, and then click **OK**.

> **Note**
> - The prefix of an endpoint must be 8 to 64 characters in length and can contain letters, digits, and hyphens (-). It must start with a lowercase letter.
> - The port number must be a value within the range of 1000 to 5999.

7.3. Apply for or release a public endpoint

ApsaraDB RDS supports two types of endpoints: internal endpoints and public endpoints. By default, you are provided with an internal endpoint that is used to connect to your ApsaraDB RDS instance. If you want to connect to your instance over the Internet, you must apply for a public endpoint.

Internal and public endpoints

Endpoint type	Description
Internal endpoint	• By default, an internal endpoint is provided. You do not need to apply for the internal endpoint. In addition, you cannot release the internal endpoint. However, you can change the network type of your instance. • If an Elastic Compute Service (ECS) instance resides in the same region and has the same network type as your RDS instance, these instances can communicate over an internal network. If your application is deployed on such an ECS instance, you do not need to apply for a public endpoint. • For security and performance purposes, we recommend that you connect to your RDS instance by using the internal endpoint.

Public endpoint	• You must manually apply for a public endpoint for your RDS instance. You can release the public endpoint if it is no longer needed.
	• If you cannot connect to your RDS instance by using the internal endpoint, you must apply for a public endpoint. You may need to apply for a public endpoint in the following scenarios:
	  ◦ You need to access an RDS instance from an ECS instance that resides in a different region or has a different network type.
	  ◦ You need to access an RDS instance from a device outside Apsara Stack.
	Note If you use a public endpoint to connect to an RDS instance, data security is compromised. Proceed with caution. For faster transmission and higher security, we recommend that you migrate your application to an ECS instance that resides in the same region and has the same network type as the RDS instance. This way, you can connect to the RDS instance by using the internal endpoint.

Apply for or release a public endpoint

1. Log on to the ApsaraDB for RDS console.
2. On the **Instances** page, find the instance that you want to manage.
3. In the left-side navigation pane, click **Database Connection**.
4. Apply for or release a public endpoint.
   - If you have not applied for a public endpoint, click **Apply for Public Endpoint**.
   - If you have applied for a public endpoint, click **Release Public Endpoint**.
5. In the message that appears, click **OK**.

FAQ

- Q: Can I change the endpoints and port numbers of my RDS instance?

  A: No, you cannot change the endpoints of your RDS instance. You can change the prefixes of the endpoints. You can also change the port numbers of your instance. For more information, see Change the endpoint and port number of an instance.

- Q: Can I configure the endpoints of my RDS instances to static IP addresses?

  A: No, you cannot configure the endpoints of your RDS instance to static IP addresses. Both primary/secondary switchovers and specification changes may cause changes to the IP addresses. Therefore, we recommend that you connect to your instance by using an endpoint. This allows you to minimize the impact on your workloads and eliminates the need to modify the configuration data on your application.

7.4. Connect to an instance

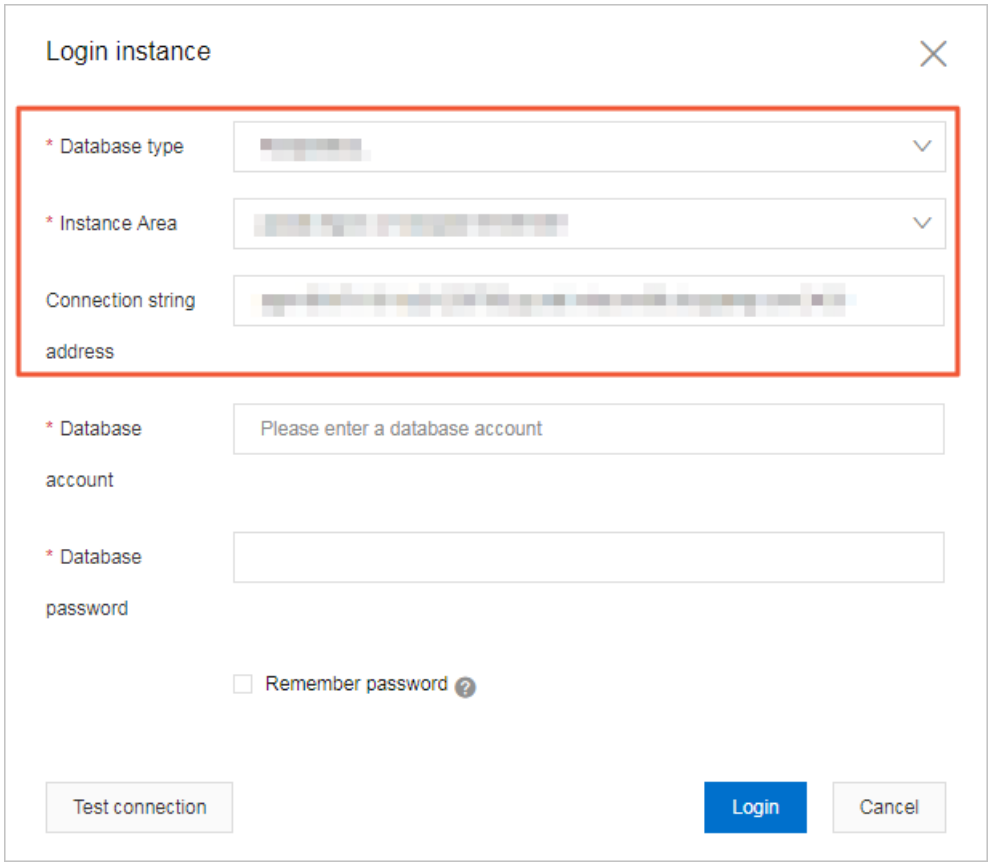
This topic describes how to use Data Management (DMS) to connect to an ApsaraDB RDS instance.

Prerequisites

- A database is created. For more information, see Create a database.
- A database account is created. For more information, see Create an account.

Procedure

1. Log on to the ApsaraDB for RDS console.
2. On the **Instances** page, find the target instance.
3. Click the instance ID or click **Manage** in the Actions column corresponding to the instance to go to the **Basic Information** page.
4. Click **Log On to DB** in the upper-right corner of the page.
5. In the **Login instance** dialog box of the **DMS** console, check values of **Database type**, **Instance Area**, and **Connection string address**. If the information is correct, enter **Database account** and **Database password**, as shown in the following figure.



Parameter	Description
Database type	The engine of the database. By default, the engine of the database to be connected is displayed.
Instance Area	The region where the instance is deployed. By default, the region of the current instance is displayed.
Connection string address	The endpoint of the instance. By default, the endpoint of the current instance is displayed.
Database account	The account of the database to be connected.
Database password	The password of the account used to connect to the database.

6. Click **Login**.

> **Note**
>
> - If you want the browser to remember the password, select **Remember password** and click **Login**.
> - If you cannot connect to the instance, check the IP address whitelist settings. For more information, see Configure a whitelist.

8.Monitoring and alerting

8.1. View resource and engine monitoring data

The ApsaraDB RDS console provides a variety of performance metrics to monitor the status of your instances.

Procedure

1. Log on to the ApsaraDB for RDS console.
2. On the **Instances** page, find the target instance.
3. Click the instance ID or click **Manage** in the Actions column corresponding to the instance to go to the **Basic Information** page.
4. In the left-side navigation pane, click **Monitoring and Alerts**.
5. On the **Monitoring and Alerts** page, select **Resource Monitoring** or **Engine Monitoring**, and select a time range to view the corresponding monitoring data. The following table describes the metrics.

Monitoring type	Metric	Description
Resource Monitoring	Disk Space (unit: MB)	The disk usage of the instance, which includes the following items: ○ Instance Size ○ Data Usage ○ Log Size ○ Temporary File Size ○ Other System File Size
	IOPS	The number of input/output operations per second (IOPS) for the instance.
	Total Connections	The total number of current connections of the instance.
	MSSQL Instance CPU Utilization (percentage in the operating system)	The CPU utilization of the instance. This includes the CPU utilization for the operating system. Unit: %.
	SQLServer Average Input/Output Traffic	The inbound and outbound traffic of the instance per second. Unit: KB.
	Average Transaction Frequency	The number of transactions processed per second.
	Average QPS	The number of SQL statements executed per second.

Monitoring type	Metric	Description
Engine Monitoring	Buffer Hit Ratio (%)	The read hit ratio of the buffer pool.
	Page Write Frequency at Check Point	The number of checkpoints written to pages per second.
	Login Frequency	The number of logons to the instance per second.
	Average Frequency of Whole Table Scans	The number of full table scans per second.
	SQL Compilations per Second	The number of SQL statements compiled per second.
	Lock Timeout Times	The number of lock timeouts on the instance per second.
	Deadlock Frequency	The number of deadlocks on the instance per second.
	Lock Wait Frequency	The number of lock waits on the instance per second.

9.Data security

9.1. Configure an IP address whitelist for an ApsaraDB RDS instance

After you create an ApsaraDB RDS instance, you must add the IP addresses or CIDR blocks that are used for database access to the IP address whitelist of the instance to ensure database security and reliability.

Context

IP address whitelists make your ApsaraDB RDS instance more secure and do not interrupt the operations of your ApsaraDB RDS instance when you configure whitelists. We recommend that you maintain your IP address whitelists on a regular basis.

Procedure

1. Log on to the ApsaraDB for RDS console.
2. On the **Instances** page, find the target instance.
3. Click the instance ID or click **Manage** in the Actions column corresponding to the instance to go to the **Basic Information** page.
4. In the left-side navigation pane, click **Data Security**.
5. On the **Whitelist Settings** tab, click **Edit** corresponding to the **default** whitelist.

   > **Note** If you want to connect an ECS instance to an ApsaraDB RDS instance by using an internal endpoint, you must make sure that the two instances are in the same region and have the same network type. Otherwise, the connection fails.

6. In the **Edit Whitelist** dialog box, enter the IP addresses or CIDR blocks that are allowed to access the instance and click **OK**.

> **Note**
>
> - Limits for IP address whitelists:
>   - You must separate multiple IP addresses with commas (,). A maximum of 1,000 different IP addresses can be added.
>
>   Supported formats are `0.0.0.0/0`, IP addresses such as `10.23.12.24`, or CIDR blocks such as `10.23.12.24/24`. `/24` indicates the length of the IP address prefix in the CIDR block. An IP address prefix can contain 1 to 32 bits.
>   - If an IP address whitelist is empty or contains only `0.0.0.0/0`, all IP addresses can access the ApsaraDB RDS instance. This may cause security risks to the instance. Proceed with caution.
> - If you click **Add Internal IP Addresses of ECS Instances**, the IP addresses of all the Elastic Compute Service (ECS) instances that are created within your Apsara Stack tenant account appear. Then, you can select the IP addresses and add them to an IP address whitelist.

9.2. Configure SSL encryption

This topic describes how to enhance endpoint security. You can enable Secure Sockets Layer (SSL) encryption and install SSL certificates that are issued by certificate authorities (CAs) to the required application services. SSL is used at the transport layer to encrypt network connections and enhance the security and integrity of communication data. However, SSL increases the response time.

Precautions

- An SSL CA certificate is valid for one year. You must update the validity period of the SSL CA certificate in your application or client within one year. Otherwise, your application or client that uses encrypted network connections cannot connect to the ApsaraDB RDS instance.
- SSL encryption may cause a significant increase in CPU utilization. We recommend that you enable SSL encryption only when you want to encrypt connections from the Internet. In most cases, connections that use an internal endpoint do not require SSL encryption.
- SSL encryption cannot be disabled after it is enabled. Proceed with caution.

Enable SSL encryption

1. Log on to the ApsaraDB for RDS console.
2. On the **Instances** page, find the target instance.
3. Click the instance ID or click **Manage** in the Actions column corresponding to the instance to go to the **Basic Information** page.
4. In the left-side navigation pane, click **Data Security**.
5. Click the **SSL Encryption** tab.
6. In the SSL Settings section, turn on **SSL Encryption**.
7. In the **Configure SSL** dialog box, select the endpoint for which you want to enable SSL encryption and click **OK**.
8. Click **Download CA Certificate** to download the SSL CA certificate files in a compressed package.

   The downloaded package contains the following files:

The downloaded package contains the following files:

- P7B file: contains the server CA certificate that can be imported into a Windows operating system.
- PEM file: contains the server CA certificate that can be imported into an operating system rather than Windows or an application that is not Windows-based.
- JKS file: contains the server CA certificate that is stored in a Java-supported truststore. You can use the file to import the CA certificate chain into a Java-based application. The default password is apsaradb.

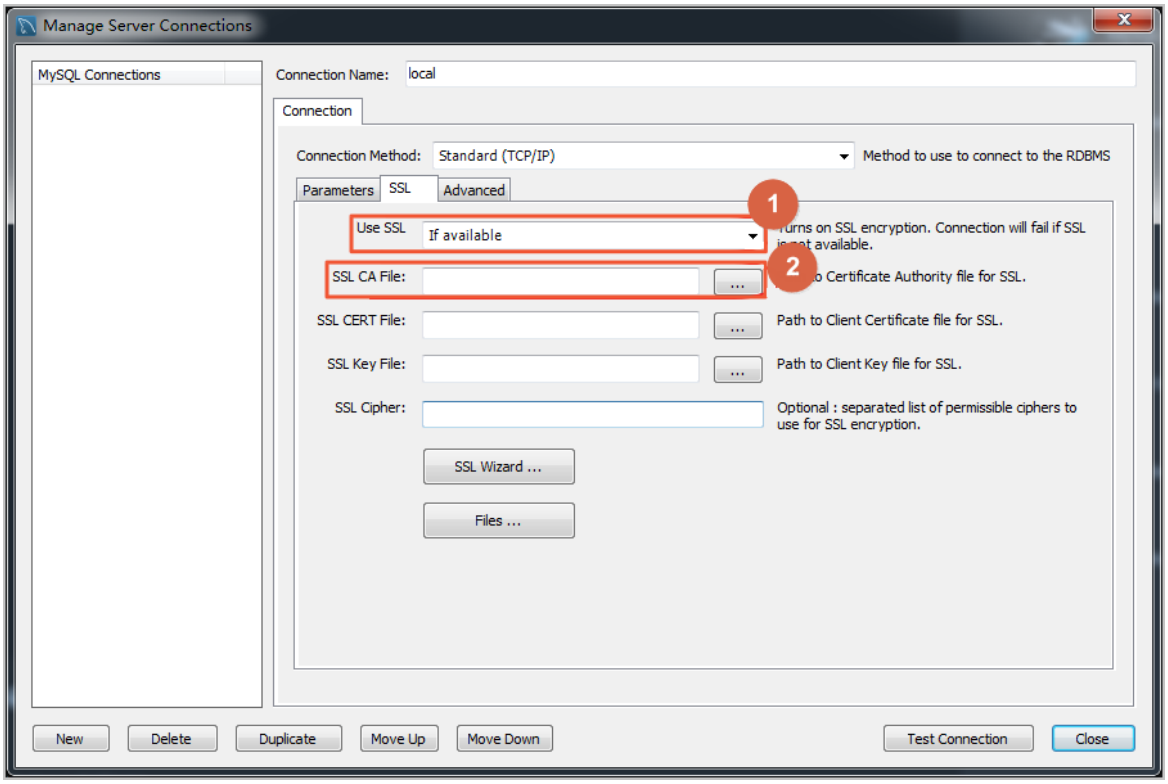
> **Note** When the JKS file is used in Java, you must modify the default JDK security configuration in JDK 7 and JDK 8. Open the `/jre/lib/security/java.security` file on the host where your application resides, and modify the following configurations:
>
> ```
> jdk.tls.disabledAlgorithms=SSLv3, RC4, DH keySize < 224
> jdk.certpath.dis
> abledAlgorithms=MD2, RSA keySize < 1024
> ```
>
> If you do not modify the JDK security configuration, the following error is reported. Similar errors are also caused by the Java security configuration.
>
> ```
> javax.net.ssl.SSLHandshakeException: DHPublicKey does not comply to algorithm con
> straints
> ```

Configure an SSL CA certificate

After you enable SSL encryption, configure the SSL CA certificate on your application or client before they can connect to the ApsaraDB RDS instance. This section describes how to configure an SSL CA certificate. MySQL Workbench and Navicat are used in the example. If you are using other applications or clients, see the related instructions.

Configure a certificate on MySQL Workbench

1. Start MySQL Workbench.
2. Choose **Database > Manage Connections**.
3. Enable **Use SSL** and import the SSL CA certificate file.



Configure a certificate on Navicat

1. Start Navicat.
2. Right-click the database and select **Edit Connection**.
3. Click the **SSL** tab. Select the path of the PEM-formatted CA certificate, as shown in the following figure.
4. Click **OK**.

> **Note** If the `connection is being used` error is reported, the previous session is still connected. Restart Navicat.

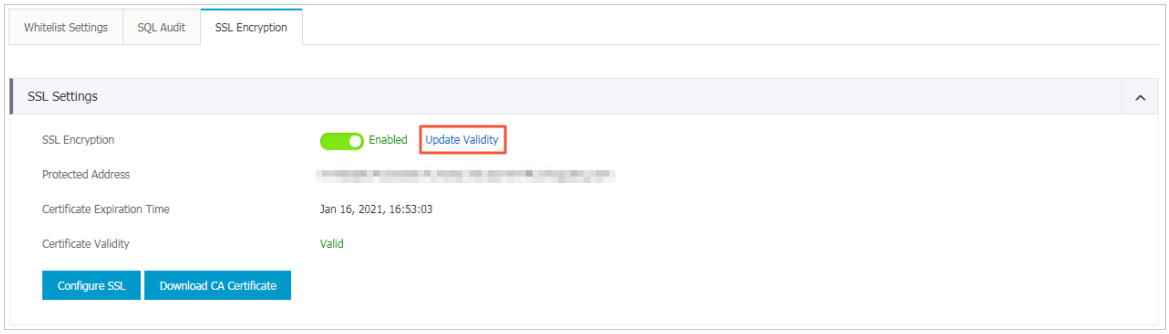
5. Double-click the database to test whether the database is connected.

Update the validity period of an SSL CA certificate

> **Note**
> - **Update Validity** causes the ApsaraDB RDS instance to restart. Proceed with caution.
> - After you perform the **Update Validity period** operation, you must download and configure the SSL CA certificate again.

1. Log on to the ApsaraDB for RDS console.
2. On the **Instances** page, find the target instance.
3. Click the instance ID or click **Manage** in the Actions column corresponding to the instance to go to the **Basic Information** page.
4. In the left-side navigation pane, click **Data Security**.

5. Click the **SSL Encryption** tab.
6. Click **Update Validity**.



7. In the message that appears, click **OK**.

9.3. Configure TDE

This topic describes how to configure Transparent Data Encryption (TDE) for your ApsaraDB RDS for SQL Server instance. TDE allows your ApsaraDB RDS instance to encrypt the data that will be written into the disk and decrypt the data that will be read from the disk to the memory. TDE does not increase the sizes of data files. When you use TDE, you do not need to modify the application that uses the ApsaraDB RDS instance.

Precautions

- Instance-level TDE can be enabled but cannot be disabled. Database-level TDE can be enabled or disabled.
- The keys used for data encryption are generated and managed by Key Management Service (KMS). ApsaraDB RDS does not provide the keys or certificates used for data encryption. If you want to restore data to your computer after TDE is enabled, you must decrypt the data on your ApsaraDB RDS instance. For more information, see Decrypt data.
- TDE increases CPU utilization.

Prerequisites

- Your ApsaraDB RDS instance runs SQL Server EE.
- KMS is activated. If KMS is not activated, you can activate it as prompted when you enable TDE.

Procedure

1. Log on to the ApsaraDB for RDS console.
2. On the **Instances** page, find the target instance.
3. Click the instance ID or click **Manage** in the Actions column corresponding to the instance to go to the **Basic Information** page.
4. In the left-side navigation pane, click **Data Security**.
5. Click the **TDE** tab.
6. Turn on **TDE Status**.
7. In the dialog box that appears, click **Confirm**.

> **Note** If you have not enabled KMS, you are prompted to do so when you enable TDE. After you enable KMS, you can turn on **TDE Status** to enable TDE.

8. Click **Configure TDE**. In the Database TDE Settings dialog box, select the databases you want to encrypt from the Unselected Databases list, click the  icon to add them to the **Selected Databases** list, and then click **OK**.

Decrypt data

If you want to decrypt a database that is encrypted by using TDE, you need only to remove the database from the Selected Databases section in the **Database TDE Settings** dialog box.

10.Service availability

10.1. Switch workloads between primary and secondary ApsaraDB RDS instances

This topic describes how to switch workloads between a primary ApsaraDB RDS instance and its secondary instance. ApsaraDB RDS supports both manual switchover and automatic switchover. After a switchover is complete, the primary RDS instance becomes the secondary instance.

Context

- Automatic switchover: By default, the automatic switchover feature is enabled. If the primary RDS instance becomes faulty, ApsaraDB RDS automatically switches workloads to the secondary instance.
- Manual switchover: You can manually switch workloads between the primary and secondary RDS instances even when the automatic switchover feature is enabled.

> **Note** Data is synchronized in real time between the primary and secondary RDS instances. You can access only the primary instance. The secondary instance serves only as a standby and does not allow external access.

Precautions

- During a switchover, your service may be interrupted. Make sure that your application is configured to automatically reconnect to the instance.
- If the primary RDS instance is attached with read-only instances, the read-only instances need to re-establish the connections that are used for data replication and synchronize incremental data after a switchover. As a result, data synchronization to the read-only instances has a latency of a few minutes.

Perform a manual primary/secondary switchover

1. Log on to the ApsaraDB for RDS console.
2. On the **Instances** page, find the instance that you want to manage.
3. In the left-side navigation pane, click **Service Availability**.
4. Click **Switch Primary/Secondary Instance** on the right side of the page.

   > **Note** You may encounter a service interruption during a switchover. Make sure that your application is configured to automatically reconnect to the instance.

5. In the dialog box that appears, click **OK**.

> **Note** In the dialog box, you can also select **Switch Within Maintenance Window** and click **OK**. Then, the system performs the primary/secondary switchover within the maintenance window. For more information about how to set the maintenance window, see Set a maintenance window. You can also click **Change** on the right to change the maintenance window.

FAQ

Q: Can I connect to secondary instances?

No, you cannot access secondary instances. You can access only the primary instance of your database system. Secondary instances serve only as a standby.

11.Database backup and restoration

11.1. Configure an automatic backup policy

Automatic backup supports full physical backups. ApsaraDB RDS automatically backs up data based on pre-configured policies. This topic describes how to configure a policy for automatic backup.

Procedure

1. Log on to the ApsaraDB for RDS console.
2. On the **Instances** page, find the target instance.
3. Click the instance ID or click **Manage** in the Actions column corresponding to the instance to go to the **Basic Information** page.
4. In the left-side navigation pane, click **Backup and Restoration**.
5. On the **Backup and Restoration** page, click the **Backup Settings** tab.
6. Click **Edit**.
7. In the dialog box that appears, configure the automatic backup policy.

Parameter	Description
Data Retention Period	The number of days for which you want to retain data backup files. Valid values: 7 to 730. Default value: 7.
Backup Cycle	The cycle based on which you want to create a backup. You can select one or more days within a week. **Note** For data security purposes, we recommend that you back up your ApsaraDB RDS instance at least twice a week.
Backup Time	The period of time for which you want to back up data. Unit: hours.
Backup Frequency	The frequency at which you want to back up logs. The following options are available: ○ Same as Data Backup ○ Every 30 Minutes The total size of log backup files remains the same regardless of the backup frequency.

8. Click **OK**.

11.2. Manually back up an instance

This topic describes how to manually back up an ApsaraDB RDS instance.

Procedure

1. Log on to the ApsaraDB for RDS console.
2. On the **Instances** page, find the target instance.
3. Click the instance ID or click **Manage** in the Actions column corresponding to the instance to go to the **Basic Information** page.
4. On the **Basic Information** page, click **Back Up Instance** in the upper-right corner.
5. In the **Back Up Instance** dialog box, select **Automatic Backup** or **Full Backup** from the **Select Backup Mode** drop-down list.

   > **Note** ApsaraDB RDS supports the following backup methods:
   >
   > - **Automatic Backup**: After you select Automatic Backup, the system immediately performs an incremental or full backup based on the instance.
   > - **Full Backup**: After you select Full Backup, the system immediately performs a full backup.

6. Click **OK**.

Result

After the backup is complete, you can view the backup task on the **Data Backup** tab of the **Backup and Restoration** page.

11.3. Shrink transaction logs

ApsaraDB RDS for SQL Server allows you to shrink transaction logs to reduce the log file size.

Prerequisites

The instance is in the **Running** state.

Procedure

1. Log on to the ApsaraDB for RDS console.
2. On the **Instances** page, find the target instance.
3. Click the instance ID or click **Manage** in the Actions column corresponding to the instance to go to the **Basic Information** page.
4. In the left-side navigation pane, click **Backup and Restoration**.
5. In the upper-right corner of the page, click **Shrink Transaction Log**. In the message that appears, click **OK**.

   > **Note** The shrinkage takes about 20 minutes to complete. ApsaraDB RDS for SQL Server shrinks transaction logs during each backup.

12.Migrate full backup data to ApsaraDB RDS for SQL Server

This topic describes how to migrate the full backup data of a self-managed SQL Server database from an Object Storage Service (OSS) bucket to an ApsaraDB RDS for SQL Server instance.

Prerequisites

- The available storage of the RDS instance is sufficient. If the available storage is insufficient, you must expand the storage capacity of the RDS instance before you start the migration.
- The names of existing databases on the RDS instance are different from the name of the self-managed database.
- A privileged account is created on the RDS instance. For more information, see Create an account.
- An OSS bucket is created in the region where the RDS instance resides. For more information, see Create buckets in the *OSS User Guide*.
- The DBCC CHECKDB statement is executed, and the return result indicates that no allocation or consistency errors occur.

  Note If no allocation or consistency errors occur, the following execution result is returned:

  ```
  ...
  CHECKDB found 0 allocation errors and 0 consistency errors in database 'xxx'.
  DBCC execution completed. If DBCC printed error messages, contact your system adminis
  trator.
  ```

Precautions

- The migration from a later SQL Server version to an earlier SQL Server version is not supported. For example, if the self-managed database runs SQL Server 2016 and the RDS instance runs SQL Server 2012, you cannot migrate the full backup data of the self-managed database to the RDS instance.
- Differential backup files and log backup files are not supported.
- The names of full backup files cannot contain special characters such as `at signs (@) and vertical bars (|)`. If the file names contain special characters, the migration fails.
- After you authorize the service account of ApsaraDB RDS to access the OSS bucket, a RAM role named **AliyunRDSImportRole** is created. Do not modify or delete this role. If you modify or delete this role, the backup files cannot be downloaded from the OSS bucket. In this case, you must re-authorize the service account by using the migration wizard.
- Before the migration is complete, do not delete the backup files from the OSS bucket. If you delete the backup files before the migration is complete, the migration fails.
- The names of the backup files can have the following suffixes: bak, diff, trn, and log. If the backup files are not generated by using the backup script that is provided in this topic, you must add one of the following suffixes to the file names:
  - bak: indicates a full backup file.
  - diff: indicates a differential backup file.
  - trn or log: indicates a transaction log backup file.

Back up the self-managed database

> **Note** Before you perform a full backup, you must stop all data writes to the self-managed database. The data that is written to the self-managed database during the full backup process cannot be backed up.

1. Download the backup script. Then, open the file by using Microsoft SQL Server Management Studio (SSMS).
2. Configure the parameters described in the following table.

Parameter	Description
@backup_databases_list	The name of the self-managed database that you want to back up. If you specify multiple databases, separate the names of these databases with semicolons (;) or commas (,).
@backup_type	The backup type. Valid values: ○ FULL: full backup ○ DIFF: differential backup ○ LOG: log backup
@backup_folder	The directory that is used to store the backup files of the self-managed database. If the specified directory does not exist, the system automatically creates one.
@is_run	Specifies whether to perform a backup or a check. Valid values: ○ 1: performs a backup. ○ 0: performs a check.

3. Run the backup script.

Upload the generated full backup file to the OSS bucket

After the full backup on the self-managed database is complete, you must use one of the following methods to upload the generated full backup file to the OSS bucket:

- Use the OSS console

  If the size of the generated full backup file is less than 5 GB, you can upload the full backup file by using the OSS console. For more information, see Upload objects in the *OSS User Guide*.

- Use the OSS API

  You can call an OSS API operation to upload the generated full backup file by using the resumable upload method. For more information, see Multipart upload-relevant operations in the *OSS Developer Guide*.

Create a migration task

1. Log on to the ApsaraDB for RDS console.
2. On the **Instances** page, find the instance that you want to manage.
3. In the left-side navigation pane, click **Backup and Restoration**.

4. In the upper-right corner of the page, click **Restore Backup Data to RDS**.
5. Click **Next** twice until the Import Data step appears.
6. Configure the parameters described in the following table.

Parameter	Description
Database Name	Enter the name of the destination database on your RDS instance. **Note** The name of the destination database must meet the requirements of open source SQL Server.
OSS Bucket	Select the OSS bucket that stores the full backup file.
OSS Subfolder Name	Enter the name of the OSS subfolder that stores the full backup file.
OSS File	Specify the full backup file that you want to import. You can enter a prefix in the search box and click the search icon to search for the full backup file by using fuzzy match. The name, size, and update time of each full backup file whose name contains the prefix are displayed. Select the full backup file that you want to migrate to the RDS instance.
Cloud Migration Method	○ **Immediate Access (Full Backup)**: If you want to migrate only a full backup file, select this migration method. In this example, select Immediate Access (Full Backup). In this case, the following parameter settings take effect in the CreateMigrateTask operation: `BackupMode = FULL` and `IsOnlineDB = True`. ○ **Access Pending (Incremental Backup)**: If you want to migrate a full backup file and a log or differential backup file, select this migration method. In this case, the following parameter settings take effect in the CreateMigrateTask operation: `BackupMode = UPDF` and `IsOnlineDB = False`.
Consistency Check Mode	○ **Asynchronous DBCC**: The DBCC CHECKDB statement is executed after the destination database is opened. This reduces the amount of time that is required to open the destination database and minimizes the downtime of your application. If the destination database is large, it may take a long time to execute the DBCC CHECKDB statement. Therefore, if your application is sensitive to downtime but is not sensitive to the result of the DBCC CHECKDB statement, we recommend that you select this consistency check mode. In this mode, the following parameter setting takes effect in the CreateMigrateTask operation: `CheckDBMode = AsyncExecuteDBCheck`. ○ **Synchronous DBCC**: The DBCC CHECKDB statement is executed at the same time when the destination database is opened. If you want to identify consistency errors between the self-managed database and the destination database based on the result of the DBCC CHECKDB statement, we recommend that you select this consistency check mode. However, the amount of time that is required to open the destination database increases. In this mode, the following parameter setting takes effect in the CreateMigrateTask operation: `CheckDBMode = SyncExecuteDBCheck`.

7. Click **OK**.

Wait until the migration task is complete. You can click **Refresh** to view the latest state of the migration task. If the migration task fails, you can troubleshoot the failure based on the description of the migration task. For more information, see Common errors.

View the migration task

If you want to view the details of the migration task, go to the **Backup and Restoration** page and click the **Backup Data Upload History** tab. By default, this tab displays the migration tasks over the last week.

Common errors

Each migration task record on the Backup Data Upload History tab of the Backup and Restoration page contains a task description. If the migration task fails or an error is reported, you can troubleshoot the failure or error based on the task description. The following common errors may occur:

- An existing database on the RDS instance has the same name as the self-managed database.
  - Error message: The database (xxx) is already exist on RDS, please backup and drop it, then try again.
  - Cause: If an existing database on the RDS instance has the same name as the self-managed database, the migration is not supported. This mechanism is designed to ensure the security of your data.
  - Solution: If you want to overwrite an existing database on the RDS instance, back up the database, delete the database from the RDS instance, and then create and run a migration task again.
- A differential backup file is used.
  - Error message: Backup set (xxx.bak) is a Database Differential backup, we only accept a FULL Backup.
  - Cause: The file that you upload is a differential backup file rather than a full backup file. The migration method in this topic supports only full backup files.
- A log backup file is used.
  - Error message: Backup set (xxx.trn) is a Transaction Log backup, we only accept a FULL Backup.
  - Cause: The file that you upload is a log backup file rather than a full backup file. The migration method in this topic supports only full backup files.
- The full backup file fails the verification.
  - Error message: Failed to verify xxx.bak, backup file was corrupted or newer edition than RDS.
  - Cause: The full backup file is corrupted, or the self-managed database runs an SQL Server version later than the RDS instance. For example, this error occurs if the self-managed database runs SQL Server 2016 and the RDS instance runs SQL Server 2012.
  - Solution: If the full backup file is corrupted, perform a full backup on the self-managed database again. Then, create and run a new migration task. If the self-managed database runs an SQL Server version later than the RDS instance, select a different RDS instance that runs the same version as or a later version than the self-managed database.
- The DBCC CHECKDB statement fails.
  - Error message: DBCC checkdb failed.

- Cause: The self-managed database encounters allocation or consistency errors.
- Solution: Execute the following statement on the self-managed database to fix the error. Then, create and run a migration task again.

  > **Note** If you execute the following statement, your data may be lost.

  ```
  DBCC CHECKDB (DBName, REPAIR_ALLOW_DATA_LOSS) WITH NO_INFOMSGS, ALL_ERRORMSGS
  ```

- The available storage of the RDS instance is insufficient.
  - Error message: Not Enough Disk Space for restoring, space left (xxx MB) < needed (xxx MB).
  - Cause: The available storage of the RDS instance is less than the minimum storage that is required to restore data by using the full backup file.
  - Solution: Expand the storage capacity of the RDS instance.
- The available storage of the RDS instance is insufficient.
  - Error message: Not Enough Disk Space, space left xxx MB < bak file xxx MB.
  - Cause: The available storage of the RDS instance is less than the size of the full backup file.
  - Solution: Expand the storage capacity of the RDS instance.
- No privileged account is created on the RDS instance.
  - Error message: Your RDS doesn't have any init account yet, please create one and grant permissions on RDS console to this migrated database (XXX).
  - Cause: No privileged account is created on the RDS instance. As a result, the system cannot determine which account needs to be authorized during the migration task. However, the data has been restored from the full backup file to the RDS instance, and the migration task is successful.
  - Solution: Create a privileged account on the RDS instance. For more information, see Create an account.