ALIBABA CLOUD

# Alibaba Cloud

## Apsara Stack Agility

Security Whitepaper

Product Version: 2009, Internal: V3.4.0

Document Version: 20210106

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.Security of Apsara Stack products

## 1.1. Object Storage Service (OSS)

### 1.1.1. Platform security

#### 1.1.1.1. Security isolation

OSS slices user data and discretely stores the sliced data in a distributed file system based on specific rules. The user data and its indexes are stored separately. OSS uses symmetric AccessKey pairs to authenticate users and verifies the signature in each HTTP request sent by users. If verification is successful, OSS reassembles the distributed data. This way, OSS implements data storage isolation between multiple tenants.

#### 1.1.1.2. Authentication and access control

##### 1.1.1.2.1. Authentication

You can create an AccessKey pair on Apsara Stack Management Console. An AccessKey pair consists of an AccessKey ID and an AccessKey secret. The AccessKey ID is a public ID that uniquely identifies a user. The AccessKey secret is private and used to authenticate a user.

Before you send a request, you must generate a signature string for the request in the format specified by OSS. Then, you must encrypt the signature string by using your AccessKey secret to generate a verification code based on the HMAC algorithm. The verification code is timestamped to prevent replay attacks. After receiving the request, OSS finds the AccessKey secret based on your AccessKey ID, and uses the AccessKey secret to decrypt the signature string and verification code. Then, OSS calculates a verification code and compares it with the decrypted verification code. If the two verification codes are the same, OSS determines that the request is valid. Otherwise, OSS rejects the request and returns HTTP 403.

##### 1.1.1.2.2. Access control

OSS supports access control list (ACL) to control access permissions. An ACL is set based on resources. You can specify ACLs for buckets or objects. You can specify an ACL for a bucket when you create the bucket or for an object when you upload the object to OSS. You can also modify the ACLs of uploaded objects and created buckets.

Access to OSS resources can be initiated by the bucket owner or third party users. An owner owns a bucket. Third-party users are other users who access resources in the bucket. Access can be either anonymous or signed. If the access is initiated with an OSS request that does not contain identification information, the access is considered to be anonymous. A signed access is a request that contains signature information in the header or a URL that contains signature information as defined in OSS API documentation.

OSS provides access control for buckets and objects.

You can configure one of the following ACLs for a bucket:

- Public read/write: All users (including anonymous users) can perform write (PutObject, GetObject, and

DeleteObject) operations on objects in the bucket.

- Public read: Only the bucket owner or authorized users can perform write operations (PutObject and DeleteObject) on objects in the bucket. Other users, including anonymous users, can only perform read operations (GetObject) from the objects in the bucket.
- Only the bucket owner or authorized users can perform read and write operations (PutObject, GetObject, and DeleteObject) on objects in the bucket. Other users cannot access the objects in the bucket without authorization.

> **Note** If you do not configure the ACL of a bucket when you create the bucket, OSS sets the ACL of the bucket to private.

You can configure one of the following ACLs for an object:

- Public read/write: All users can perform read/write operations on the object.
- Public read: Only the object owner can perform read/write operations on the object. Others can perform read operations on the object.
- Private: Only the object owner can perform read/write operations on the object. Others cannot access the object.
- Default: The object inherits the ACL of the bucket.

> **Note** If you do not configure the ACL of an object when you upload the object, OSS sets the ACL of the object to default.

# 1.1.1.2.3. Support for RAM and STS

OSS supports Resource Access Management (RAM) and Security Token Service (STS) authentication.

RAM is a resource access control service provided by Apsara Stack. RAM allows you to create RAM users under an Apsara Stack tenant account. The Apsara Stack tenant account can grant access permissions on resources to RAM users.

STS is service that provides temporary access credentials. You can use STS to generate a temporary access credential for a user and specify the permission and validity period of the credential. A credential becomes invalid after it expires.

# 1.1.1.3. Data security

An error may occur when data is transferred between the client and server. OSS supports CRC and MD5 verification to secure data.

## CRC

OSS can return the CRC64 value of objects uploaded through any of the methods provided. The client can compare the CRC64 value with the locally calculated value to verify data integrity.

OSS calculates the CRC64 value for newly uploaded objects and stores the result as metadata of the object. OSS then adds the x-oss-hash-crc64ecma header to the returned response header, indicating its CRC64 value. This CRC64 value is calculated based on Standard ECMA-182.

## MD5 verification

To check whether the object uploaded to OSS is consistent with the local file, attach the Content-MD5 field value to the upload request. The OSS server verifies the MD5 value. The upload can succeed only when the MD5 value of the object received by the OSS server is the same as the Content-MD5 field value. This method can ensure the consistency between objects.

# 1.1.1.4. Data encryption

## 1.1.1.4.1. Server-side encryption

OSS supports server-side encryption for uploaded data. When you upload data, OSS encrypts the data by using AES256 and permanently stores the encrypted data. When you download the data, OSS automatically decrypts the data, returns the original data, and declares in the header of the returned HTTP request that the data had been encrypted on the server.

To encrypt an object on the OSS server when you upload the object, you only need to add the x-oss-server-side-encryption header in the PutObject request and set its value to AES256.

## 1.1.1.4.2. Client-side encryption

OSS allows you to use client-side encryption to encrypt data before the data is sent to the server while the data encryption key (DEK) used is kept only on the local client. Other users cannot obtain the raw data without the DEK and enveloped data key (EDK), even if the data is leaked. OSS uses functions provided by SDKs to encrypt the data on local clients before the data is uploaded to the OSS bucket.

# 1.1.2. Tenant security

## 1.1.2.1. Log audit

OSS automatically saves access logs. After access logging is enabled for a source bucket, OSS generates an object that contains access logs for that bucket (by hour), names the object based on predefined naming rules, and writes the object into the bucket specified by the user. These logs are used for later auditing and behavior analysis. Request logs contain information such as the request time, source IP address, request object, return code, and processing duration.

## 1.1.2.2. Key management

Apsara Stack Key Management Service (KMS) is a secure and highly available service that integrates hardware and software, and provides a key management system that can be extended to the cloud. KMS uses customer master keys (CMKs) to encrypt OSS objects and uses KMS API operations to generate data encryption keys (DEKs) in a centralized manner. You can define policies in KMS to control and monitor key usage. You can use these keys to protect data in OSS buckets.

## 1.1.2.3. Configure hotlink protection

To prevent additional fees caused by unauthorized access to the resources in your bucket, you can configure hotlink protection for your buckets on the OSS console or by using API operations.

You can set the following parameters to configure hotlink protection:

- Referer Whitelist: Only specified domain names are allowed to access OSS resources.
- Allow Empty Referer: If this parameter is disabled, a request is allowed to access OSS resources only

if the request includes the Referer field configured in the HTTP or HTTPS header.

For example, for a bucket named oss-example, you can add http://www.aliyun.com/ to the Referer whitelist. Requests in which the Referer field is http://www.aliyun.com/ can access the objects in this bucket.

# 1.2. ApsaraDB for RDS

## 1.2.1. Platform security

### 1.2.1.1. Secure isolation

#### Tenant isolation

ApsaraDB for RDS uses virtualization technology to isolate tenants. Each tenant can maintain their own database permissions independently. Alibaba Cloud also implements increased security for servers that run databases to prevent other users from accessing your data. For example, databases cannot read or write system files.

### 1.2.1.2. Authentication

ApsaraDB for RDS secures data through authentication.

#### Identity authentication

Account authentication uses your logon password or AccessKey pair to verify your identity. You can create an AccessKey pair from Apsara Stack Management Console. An AccessKey pair consists of AccessKey ID and AccessKey Secret. AccessKey ID is a public key used for identification. AccessKey Secret is used to encrypt signature strings sent from the client and verify signature strings sent by the server. You must keep your AccessKey Secret confidential.

The ApsaraDB for RDS server authenticates the sender identity of each access request. Because of this, each request must contain signature information, regardless of whether it is sent using HTTP or HTTPS. ApsaraDB for RDS uses AccessKey ID and AccessKey Secret to implement symmetric-key encryption and authenticate the identity of a request sender. AccessKey pairs can be applied for and managed from the Apsara Stack. The AccessKey Secret will only be known to you, so it is necessary to take precautions to keep it confidential.

#### Permission control

ApsaraDB for RDS does not automatically create initial database accounts for a newly created instance. You can use the console or API to create a standard database account and configure database-level read and write permissions. To implement fine-grained permission control, such as table-level, view-level, or field-level permissions, you can use the console or API to create a master database account. You can then use the database client and master database account to create standard database accounts. A master database account can configure table-level read/write permissions for standard database accounts.

#### Access control

All ApsaraDB for RDS instances that are created by an Apsara Stack tenant account are managed as resources by that account. By default, an Apsara Stack tenant account is granted full operation permissions on all resources belonging to the account.

ApsaraDB for RDS supports Resource Access Management (RAM). You can use RAM to allow RAM users to access and manage RDS resources under your account. ApsaraDB for RDS can also provide short-term access permissions with temporary credentials provided through STS.

# 1.2.1.3. Data security

ApsaraDB for RDS secures data through hot standby, data backups, and log backups.

High-availability ApsaraDB for RDS instances implement two database nodes for hot standby. When the primary node fails, the secondary node immediately takes over services. Database backups can be initiated anytime. To improve data traceability, ApsaraDB for RDS can restore data to any previous point in time based on the backup policy.

Automatic backup at regular intervals is required to guarantee the integrity, reliability, and restorability of databases. ApsaraDB for RDS provides two backup functions: data backup and log backup.

# 1.2.1.4. Data encryption

## SSL

ApsaraDB for RDS provides Secure Sockets Layer (SSL) for MySQL and SQL Server. You can prevent man-in-the-middle attacks by using the server root certificate to verify whether the destination database service is provided by RDS. RDS also allows you to enable and update SSL certificates for servers to guarantee security and validity.

Although ApsaraDB for RDS can encrypt the connection between an application and a database, SSL cannot run properly until the application authenticates the server. SSL consumes extra CPU resources and affects the throughput and response time of instances. The severity of the impact depends on the number of user connections and frequency of data transfers.

# 1.2.1.5. DDoS attack prevention

ApsaraDB for RDS prevents DDoS attacks by using the traffic scrubbing and black hole filtering features.

When you access an ApsaraDB for RDS instance from the Internet, the instance is vulnerable to DDoS attacks. When a DDoS attack is detected, the RDS security system first scrubs inbound traffic. If traffic scrubbing is insufficient or if the black hole threshold is reached, black hole filtering is triggered.

Triggering conditions for traffic scrubbing and black hole filtering are listed as follows:

- **Traffic scrubbing**

  Traffic scrubbing only targets traffic from the Internet. Traffic is redirected from an IP address to the scrubbing device, which then checks whether the traffic is normal. Abnormal traffic is discarded and traffic to the server is limited by the scrubbing device to mitigate damage on the server. These operations may have an impact on normal traffic.

  ApsaraDB for RDS triggers and stops traffic scrubbing automatically. Traffic scrubbing is triggered for a single ApsaraDB for RDS instance if any of the following conditions are met:

  - Packets per second (PPS) reaches 30,000.
  - Bits per second (BPS) reaches 180 Mbit/s.

- The number of new concurrent connections per second reaches 10,000.
- The number of active concurrent connections reaches 10,000.
- The number of inactive concurrent connections reaches 10,000.

- **Black hole filtering**

  Black hole filtering only targets traffic from the Internet. If an RDS instance is undergoing black hole filtering, the instance cannot be accessed from the Internet and connected applications will not be available. Black hole filtering is triggered for a single ApsaraDB for RDS instance if any of the following conditions are met:

  - BPS reaches 2 Gbit/s.
  - Traffic scrubbing is ineffective.

  Black hole filtering is automatically stopped 2.5 hours after being triggered. Then, the instance will undergo traffic scrubbing. If the DDoS attack is still occurring, black hole filtering is triggered again. Otherwise, the system restores the normal state.

# 1.2.2. Tenant security

## 1.2.2.1. Log audit

ApsaraDB for RDS can audit logs to identify security issues.

ApsaraDB for RDS allows you to view SQL transactions and periodically audit the SQL server to identify and resolve issues. RDS Proxy records all SQL statements sent to ApsaraDB for RDS, including the IP address, database name, user account used for execution, execution period, number of returned records, and execution time of each statement.

## 1.2.2.2. IP address whitelist

ApsaraDB for RDS uses the IP address whitelist to prevent access from invalid IP addresses.

ApsaraDB for RDS instances can be accessed from any IP address by default. Because of this, the IP address whitelist contains only the entry 0.0.0.0/0. You can add IP address whitelist rules through the data security module in the console or by calling an API. The IP address can be updated without restarting the ApsaraDB for RDS instance. Whitelist updates will not affect the normal operation of the instance. Multiple groups can be configured in the IP address whitelist. Each group can contain up to 1,000 IP addresses or IP address segments.

## 1.2.2.3. Software update

ApsaraDB for RDS supports post-restart update and mandatory update for software.

ApsaraDB for RDS automatically provides you with new versions of installed database software. In most cases, it is not required to update software immediately. Only when you manually restart an ApsaraDB for RDS instance does the system update the database software to the latest compatible version.

In rare cases such as critical bugs and security vulnerabilities, ApsaraDB for RDS will force the database to update during the maintenance period of the instance. Such mandatory updates only result in temporary database disconnections, and will not have any adverse impact on the application if the database connection pool is configured properly.

You can use the console or API to change the maintenance schedule to prevent a mandatory update from occurring during peak hours.

# 1.3. Data Transmission Service (DTS)

## 1.3.1. Platform security

### 1.3.1.1. Security isolation

DTS uses independent processes and files to isolate instances and data between tenants. For example, users are not allowed to read/write OS files of instances so that users cannot access data of other users.

### 1.3.1.2. Authentication

You can use your Alibaba Cloud account to create a DTS instance. The resources of the DTS instance are owned by the Alibaba Cloud account. The account has full access permissions on its DTS resources by default.

DTS supports RAM for Alibaba Cloud. You can assign permissions to access and manage DTS resources to RAM users. RAM enables you to assign permissions as needed and helps enterprises minimize information security risks.

### 1.3.1.3. Transmission security

To enhance data transmission security, DTS-defined log formats are used.

In DTS, data is encrypted for secure transmission. For example, data is encrypted during incremental data synchronization between the data reading module and the data synchronization module.

DTS also supports HTTPS to effectively improve access security.

### 1.3.1.4. Data security

When you use Data Transmission Service (DTS) to synchronize or subscribe to incremental data, the data is stored on DTS servers. The data is serialized and stored based on the storage format that is defined in DTS. The DTS-defined storage format provides enhanced data security.

> **Note** After data is stored for seven days, the data is automatically cleared.

# 1.4. Cloud Native Distributed Database PolarDB-X

## 1.4.1. Platform security

### 1.4.1.1. Security isolation

#### Network isolation

PolarDB-X supports advanced control of network access by using a Virtual Private Cloud (VPC).

A VPC is a private network environment that you set. It strictly isolates network packets through underlying network protocols, and it controls access at the network layer. The VPC and IP address whitelist together greatly improve the security of PolarDB-X instances.

## 1.4.1.2. Authentication

PolarDB-X provides a system to manage accounts and permissions, similar to that of MySQL. This system supports commands and functions such as GRANT, REVOKE, SHOW GRANTS, CREATE USER, DROP USER, and SET PASSWORD.

When you create a PolarDB-X database, by default, you can specify an account with all permissions. You can use this account to create one or more new accounts.

- You can grant permissions at the database and table levels. Currently, global permissions and column-level permissions are not supported.
- These eight statements of associated basic permissions are supported: CREATE, DROP, ALTER, INDEX, INSERT, DELETE, UPDATE, and SELECT.
- You can use `user@'host'` to match and verify access to a host.

> **Note** However, if the business host is in a Virtual Private Cloud (VPC) network, the IP address cannot be obtained due to technical restrictions. In this case, we recommend that you change the format to `user@'%'`.

# 1.4.2. Tenant security

## 1.4.2.1. IP address whitelist

PolarDB-X provides IP address whitelists to ensure secure access. You can configure an IP address whitelist for each PolarDB-X database.

The default setting of PolarDB-X instances allows access from any IP address. You can add IP addresses to the whitelist on the **Whitelist Settings** page in the console. You are required to restart PolarDB-X instance after you update the IP address whitelist, and your operations on the instance are not affected. You can set IP addresses or CIDR blocks in the IP address whitelist.

> **Note** If the business host is in a Virtual Private Cloud (VPC) network, the IP address cannot be obtained due to technical restrictions. We recommend that you remove the IP address whitelist.

## 1.4.2.2. Protection against high-risk SQL operations

PolarDB-X prohibits high-risk operations such as full table deletion and full table update by default. You can temporarily skip this restriction by adding a hint. The following statements are prohibited by default:

- DELETE statements that do not contain the WHERE or LIMIT conditions.
- UPDATE statements that do not contain the WHERE or LIMIT conditions.

For example, the following statement is prohibited:

```
mysql> delete from tt;
ERR-CODE: [TDDL-4620][ERR_FORBID_EXECUTE_DML_ALL] Forbid execute DELETE ALL or UPDATE ALL sql. M
ore: [http://middleware.alibaba-inc.com/faq/faqByFaqCode.html?faqCode=TDDL-4620]
```

After a hint is added, the statement is successfully executed.

```
mysql> /*TDDL:FORBID_EXECUTE_DML_ALL=false*/delete from tt;
Query OK, 10 row affected (0.21 sec)
```

## 1.4.2.3. Slow SQL audit

In the PolarDB-X console, you can query the slow SQL statements sent by an application to PolarDB-X. Slow SQL statements increase the response time (RT) of the entire link and reduce the throughput of PolarDB-X.

Contents of a slow SQL statement include the execution start time, database name, SQL statement, client IP address, and execution time. You can query details of slow SQL statements in the PolarDB-X console for optimization and adjustment.

## 1.4.2.4. Performance monitoring

The PolarDB-X console provides monitoring metrics in different dimensions. You can perform related operations based on the monitoring information.

There are two types of PolarDB-X monitoring information:

- Monitoring information about resources, including the CPU, memory, and network.
- Monitoring information about engines, including the logical queries per second (QPS), physical QPS, logical response time (RT) in milliseconds, physical RT in milliseconds, number of connections, and number of active threads.

The QPS and CPU performance of a PolarDB-X instance are in positive correlation. When PolarDB-X encounters a performance bottleneck, the CPU utilization of the PolarDB-X instance remains high. If the CPU utilization exceeds 90% or remains above 80%, the PolarDB-X instance faces a performance bottleneck. If there is no bottleneck in the PolarDB-X instance, the curent type of the PolarDB-Xinstance cannot meet the QPS performance requirements of the business. In this case, upgrade the instance.