Copyright © Huawei Technologies Co., Ltd. 2020. All rights reserved.
No part of this document may be reproduced or transmitted in any form or by any means without prior
written consent of Huawei Technologies Co., Ltd.

Trademarks and Permissions

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

Notice

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

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

1 Overview of OBS Best Practices ................................................................. 1

2 Migrating Local Data to OBS ................................................................. 3
  2.1 Overview ............................................................................................ 3
  2.2 Migrating Through OBS Tools .......................................................... 5
  2.3 Migrating Through the CDM Service ............................................... 5
  2.4 Migrating Through Disk-Based DES .............................................. 6
  2.5 Migrating Through Teleport-Based DES ......................................... 7
  2.6 Migrating Through Direct Connect .................................................. 8

3 Using Backup Software to Back Up Local Data to OBS ...................... 9
  3.1 Overview ............................................................................................ 9
  3.2 Using Commvault to Back Up Local Data in SAP HANA ................. 9

4 Accessing OBS over Intranet ................................................................. 12
  4.1 Overview ............................................................................................ 12
  4.2 Accessing OBS over Intranet by Using OBS Browser+ on a Windows ECS ............................................................................. 14
  4.3 Accessing OBS over Intranet by Using obsutil on a Linux ECS ........ 17

5 Using a User-Defined Domain Name to Host a Static Website ........... 20
  5.1 Overview ............................................................................................ 20
  5.2 Static Website Hosting ........................................................................ 21
  5.3 Updating a Static Website ................................................................... 26

6 Enterprise Data Access Control ............................................................. 29
  6.1 Introduction to OBS Access Control ................................................ 29
  6.2 Access Management on Department Public Data ............................ 36
  6.3 Data Sharing Among Departments/Projects .................................... 38
  6.4 Data Isolation from Enterprise Partners .......................................... 44
  6.5 Authorizing Business Departments with Independent Resource Permissions ................................................................. 45
  6.6 Isolating Bucket Resources Between Business Departments .......... 51

7 Verifying Data Consistency ................................................................. 60
  7.1 Overview ............................................................................................ 60
  7.2 Verifying Data Consistency During Upload .................................... 63
  7.3 Verifying Data Consistency During Download ................................. 66

8 Performance Optimization ................................................................. 70
9 Using a Presigned URL to Access OBS.......................... 71
A Change History......................................................... 76
1 Overview of OBS Best Practices

This document summarizes practices in common application scenarios of Object Storage Service (OBS). Each practice case provides detailed solution description and operation guide, helping you easily build your storage services based on OBS.

Table 1-1 OBS best practices

<table>
<thead>
<tr>
<th>Best Practice</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Migrating Local Data to OBS</td>
<td>This section describes how to migrate local data from personal computers or on-premises storage servers to OBS.</td>
</tr>
<tr>
<td>Using Backup Software to Back Up Local Data to OBS</td>
<td>This section describes the backgrounds of backing up local data to OBS and the backup software supported by OBS. Commvault is used as an example to describe how to back up local data to OBS.</td>
</tr>
<tr>
<td>Accessing OBS over Intranet</td>
<td>This section describes how to access OBS from your Elastic Cloud Servers (ECSs) through the internet or the HUAWEI CLOUD intranet. To optimize performance and reduce costs, it is recommended that you access OBS over the HUAWEI CLOUD intranet. ECS supports access to OBS through either Internet or the HUAWEI CLOUD intranet.</td>
</tr>
<tr>
<td>Enterprise Data Access Control</td>
<td>OBS provides multiple permission control mechanisms to help you manage data stored in OBS. This section uses common scenarios of data permission control as examples to describe how to control access to data stored in OBS to ensure data security.</td>
</tr>
<tr>
<td>Performance Optimization</td>
<td>This section describes how to add random prefixes to object names to implement horizontal expansion for access requests, and thus improve the access rate and shorten the access delay.</td>
</tr>
<tr>
<td>Best Practice</td>
<td>Description</td>
</tr>
<tr>
<td>----------------------------------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Using a Presigned URL to Access OBS</td>
<td>This section describes how to use presigned URLs to access OBS from applications developed based on OBS. This method can better protect application data and prevent data leakage and unauthorized access in case of being attacked.</td>
</tr>
</tbody>
</table>
2 Migrating Local Data to OBS

2.1 Overview

Background

Conventional on-premises storage servers cannot meet the storage demands for massive amounts of data. The main reasons are as follows:

- Storage capacity is subject to hardware devices. If the storage capacity becomes insufficient, you need to purchase disks and expand the capacity manually.
- The initial deployment requires high investment and long construction period, but it quickly lags behind as enterprise services change so fast.
- Network information vulnerabilities, technical vulnerabilities, and misoperations may result in unbearable security risks.

In contrast, OBS provides massive, stable, and secure cloud storage capabilities. With OBS, you do not need to worry about the storage capacity because it can be expanded infinitely. OBS can store unstructured data of any type and size. OBS ensures high stability and security for your data, featuring a multi-level reliability architecture, server-side encryption, log management, and permission control. In terms of the cost, OBS is available upon service subscription, eliminating the need for the investment in physical server deployment and maintenance.

HUAWEI CLOUD provides migration solutions to help you migrate data from your on-premises storage servers to OBS in a cost-effective, secure, and efficient manner. You can select a suitable migration solution according to your data volume, time arrangement, and budget.

Migration Solutions

Table 2-1 describes the migration solutions provided by HUAWEI CLOUD.
### Table 2-1 Migration solutions

<table>
<thead>
<tr>
<th>Migration Method</th>
<th>Data Volume</th>
<th>Requirement</th>
<th>Time Required</th>
<th>Pricing</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Migrating Through OBS Tools (online)</strong></td>
<td>Not larger than 1 TB</td>
<td>Sufficient public network bandwidth; requiring manual operations on clients or scripts to start data upload.</td>
<td>About 1 day for 1 TB data with the bandwidth of 100 Mbit/s</td>
<td>Data transmission is offered for free. Fees are charged only for storage space used on OBS.</td>
</tr>
<tr>
<td><strong>Migrating Through the CDM Service (online)</strong></td>
<td>Less than 8 TB at a time</td>
<td>Subscription to the CDM service is required.</td>
<td>1 TB to 8 TB each day (depending on the network condition and the read and write performance of the data source)</td>
<td>Fees are charged based on CDM instance specification and the running duration. For details, see CDM Pricing Details.</td>
</tr>
<tr>
<td><strong>Migrating Through Disk-Based DES (offline)</strong></td>
<td>Less than 10 TB at a time</td>
<td>Disks need to be prepared.</td>
<td>Although it takes 1 to 2 days to deliver disks to a data center, 4 TB data can be imported within one day.</td>
<td>Fees are charged based on the number of disks and the running duration. For details, see DES Pricing Details.</td>
</tr>
<tr>
<td><strong>Migrating Through Teleport-Based DES (offline)</strong></td>
<td>Less than 100 TB at a time</td>
<td>Huawei data centers provide Teleports for data transmission.</td>
<td>Although it takes 1 to 2 days to deliver the Teleport to a data center, 120 TB data can be imported within two days.</td>
<td>Fees are charged based on the number of disks and the running duration. For details, see DES Pricing Details.</td>
</tr>
</tbody>
</table>
### Migration Method

<table>
<thead>
<tr>
<th>Migration Method</th>
<th>Data Volume</th>
<th>Requirement</th>
<th>Time Required</th>
<th>Pricing</th>
</tr>
</thead>
<tbody>
<tr>
<td>Migrating Through Direct Connect (real-time)</td>
<td>More than 100 TB data that needs real-time online transmission every month</td>
<td>Private lines need to be deployed.</td>
<td>Depends on the bandwidth of the private line.</td>
<td>Fees are charged based on the distance and bandwidth of the private line. For details, see Direct Connect Pricing Details.</td>
</tr>
</tbody>
</table>

2.2 Migrating Through OBS Tools

OBS tools are suitable for data migration within the hundreds of gigabytes. OBS provides various client tools, such as OBS Browser+ and obsutil, to help you migrate data from local hosts to OBS. Uploading data occupies your public network bandwidth. Therefore, you are advised to upload data during off-peak hours.

For details about the usage scenarios and operation guide of each tool, see OBS Tools Guide.

2.3 Migrating Through the CDM Service

Overview

Cloud Data Migration (CDM) provides batch data migration services between homogeneous and heterogeneous data sources. By creating scheduled jobs, CDM connects data sources, such as file systems, databases, and object storage on the on-premises storage servers, to HUAWEI CLOUD OBS. In this way, local data can be migrated to OBS periodically and automatically.
1. Creating an OBS bucket
   Create a bucket on OBS Console or OBS Browser+ for storing data.

2. Purchasing CDM
   Purchase the CDM service, that is, create a CDM cluster to manage connections and jobs.

3. Configuring connections and jobs
   Create a source connection and a destination connection in the created CDM cluster to respectively connect to the local data source and OBS in the cloud. Then create a CDM job to migrate local data to OBS.

4. Starting data transmission
   Run the CDM job to start data transmission. You can view the job progress on the job management page.

**Example: Periodically Backing Up Files from an FTP Server to OBS**

CDM can periodically upload new files to OBS. You do not need to compile code or manually upload the files. You can also use the massive storage capabilities of OBS on HUAWEI CLOUD to back up files. For details about how to periodically back up files from an FTP server to OBS, see From FTP/SFTP to OBS.

**2.4 Migrating Through Disk-Based DES**

Disk-based Data Express Service (DES) allows users to deliver data disks (such as USB flash drives and eSATA disks) to a data center of HUAWEI CLOUD, achieving efficient data transmission. Disk-based DES is suitable for TB-scale data migration.
2.5 Migrating Through Teleport-Based DES

Teleport is a storage device specially designed for TB- or PB-scale data migration to OBS. It is dust- and water-proof and resistant to vibration and crush. With multiple security protection mechanisms, such as GPS locking, data encryption, and offline transfer, Teleport ensures data security and can efficiently complete large-scale data migration.

1. Creating an OBS bucket
Create a bucket on OBS Console or OBS Browser+ for storing data.

For details, see Detailed Instructions on Using Disks.
2. Creating a Teleport-based DES order

DES provides Teleport-based and disk-based services. Select Teleport-based DES in this scenario.

3. Receiving and importing data to the Teleport

After the DES order is created successfully, you will receive the Teleport sent by a HUAWEI CLOUD data center. Perform simple configuration to connect the Teleport to your data server, copy the data, and send the Teleport to the HUAWEI CLOUD data center.

4. Starting data transmission

After the HUAWEI CLOUD data center receives the Teleport, you can input the access keys on DES Console to transmit data from the Teleport to a specified OBS bucket. After data transmission is complete, you can view the transmission result on DES Console or OBS Console.

For details, see Detailed Instructions on Using Teleport.

2.6 Migrating Through Direct Connect

Direct Connect connects your data center to HUAWEI CLOUD, so that you can upload local data directly to HUAWEI CLOUD OBS. Direct Connect is recommended when local data needs to be migrated to OBS frequently or in real time. The provided low-latency and high-bandwidth services facilitate uploading data to OBS at any time.

![Data migration diagram of Direct Connect](image)

1. Creating an OBS bucket

Log in to OBS Console and create one or more buckets for storing user data.

2. Enabling Direct Connect

Log in to Direct Connect Console, fill in the application form and submit an order. After the administrator approves the application, you can pay for the order and contact the carrier for physical line connections. Huawei engineers will cooperate with your carrier to configure the connection. For details, see Process Description.

3. Starting data transmission

After Direct Connect is enabled, you can upload local data to OBS using the management console, tool, APIs, or SDKs.
3 Using Backup Software to Back Up Local Data to OBS

3.1 Overview

In traditional backup and restoration solutions, backup data needs to be written to storage devices such as tapes and then transported to a data center. In this process, data security and integrity are subject to many factors, such as hardware performance and persons. In addition, data center deployment and maintenance pose problems such as complex management and high costs.

Cloud storage is easy-to-use, secure, efficient, and cost-effective, making it an attractive substitute for traditional storage devices such as tapes. OBS is a cloud storage service that provides massive and scalable storage services. All OBS services and storage nodes work in distributed cluster mode to improve OBS scalability. Data redundancy and consistency check functions improve the security and reliability of data stored on OBS. Owing to OBS's pay-per-use billing mode, your cost on OBS is easy to estimate.

Backup software, such as Commvault and AnyBackup Cloud, can be connected to OBS for data backup. With such backup software, you can customize backup policies based on your requirements to achieve secure and efficient backup.

3.2 Using Commvault to Back Up Local Data in SAP HANA

SAP HANA is a high-performance real-time data computing platform based on the memory computing technology. Enterprises that need to process a large amount of real-time service data may use SAP HANA. The backup software Commvault is seamlessly integrated with SAP HANA and OBS and supports backup for online databases and logs. When a fault occurs in the SAP HANA system or service migration is required, Commvault can help you quickly and easily restore data, thereby providing enterprise-level data protection for SAP HANA.
Commvault V11 is recommended in this scenario.

Logical Architecture

The following uses Commvault as an example to describe how to back up the SAP HANA deployed on a local single-node system. Figure 3-1 shows the logical architecture.

Figure 3-1 Logical architecture

Table 3-1 describes the components in the logical architecture.

Table 3-1 Component description

<table>
<thead>
<tr>
<th>Component</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>iDataAgent (iDA)</td>
<td>Backup client agent, which is deployed on the SAP HANA node to obtain data to be backed up from SAP HANA.</td>
</tr>
<tr>
<td>CommServe (CS)</td>
<td>Backup server, which is deployed on the backup management node and is responsible for formulating global backup policies and scheduling backup services.</td>
</tr>
<tr>
<td>MediaAgent (MA)</td>
<td>Backup media, which is deployed on the backup service node and stores backup data to OBS.</td>
</tr>
<tr>
<td>OBS</td>
<td>In backup scenarios, OBS stores backup data. Buckets are containers on OBS and data is stored in OBS buckets.</td>
</tr>
</tbody>
</table>

NOTE

A CommCell is a backup management domain and a logical grouping of software components. Such software components obtain, transmit, store, and manage data and information.
Backup Process

1. Installing and pre-configuring the backup software
   When backing up SAP HANA, you need to install and configure the backup server (CommServer), backup media (MediaAgent), and SAP HANA backup client agent (iDataAgent).

2. Creating backup storage space (OBS bucket)
   a. Log in to OBS Console and create a bucket as the backup data storage space. For details about how to create a bucket, see Creating a Bucket.
   b. Create a cloud repository on CommCell Console. Enter the OBS endpoint address, access keys, and the bucket name to associate the MediaAgent of Commvault with OBS.

   ☐ NOTE
   CommCell Console is a graphical user interface for managing CommCell environments, monitoring and controlling activity jobs, and viewing activity-related events.

3. Creating a Commvault backup policy
   Create a backup policy on Commcell Console and specify the backup period, time, and encryption mode.

4. Checking the backup execution status
   During the execution of a backup policy, you can view the backup execution status on Commcell Console.

5. (Optional) Restoring data
   Restore data to the SAP HANA source host.

   ☐ NOTE
   For details about Commvault operations, see Commvault Official Documentation.
4 Accessing OBS over Intranet

4.1 Overview

Scenario Introduction

An enterprise runs basic services on Elastic Cloud Servers (ECSs), but storage capacity of hard disks becomes insufficient for storing a large number of images and videos. After learning that HUAWEI CLOUD provides OBS, an elastic cloud storage service for massive amounts of data, the enterprise determined to use OBS as the data storage resource pool to reduce the burden on local servers.

From ECSs, you can access OBS over the internet or HUAWEI CLOUD intranet. However, for access over the internet, the network response speed is subject to the network conditions, and you need to pay for data access over the internet. To maximize performance and reduce costs, enterprise administrators want to access OBS over the intranet.

NOTE

When accessing OBS over the intranet, ensure that the OBS resources to be accessed are in the region where the ECS resides. If the OBS resources reside in a different region, access is supported only over the Internet.

Solution

Configure intranet DNS on the established ECS. The intranet DNS resolves the OBS domain name so that the ECS can access OBS through the intranet. Figure 4-1 shows the access process.
Table 4-1 describes the services in the figure.

Table 4-1 Service description

<table>
<thead>
<tr>
<th>Service</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Virtual Private Cloud (VPC)</td>
<td>VPC enables users to create an isolated virtual network environment defined and managed by themselves, improving security of resources in the cloud and simplifying network deployment. A subnet is a network that provides IP address management and DNS services for the ECS in a VPC. IP addresses of an ECS must be in the same subnet of the ECS.</td>
</tr>
<tr>
<td>Domain Name Service (DNS)</td>
<td>Intranet DNS is provided for resolving intranet domain names and OBS domain names. This simplifies the domain name resolution process and reduces costs on data transfer over the internet.</td>
</tr>
</tbody>
</table>

- For Windows ECSs, you are advised to use OBS Browser+ to access OBS over intranet. For details, see:
  - **Accessing OBS over Intranet by Using OBS Browser+ on a Windows ECS**
- For Linux ECSs, you are advised to use obsutil to access OBS over intranet. For details, see:
  - **Accessing OBS over Intranet by Using obsutil on a Linux ECS**

When accessing OBS through the intranet from your ECSs, you can read, back up, and archive data without affecting the internet bandwidth.
4.2 Accessing OBS over Intranet by Using OBS Browser+ on a Windows ECS

OBS Browser+ is a graphical interface tool for managing OBS resources. You can configure the intranet DNS server address to access OBS over intranet from a HUAWEI CLOUD Windows ECS. The process and procedure are described as follows:

**Process**

**Figure 4-2** The process of accessing OBS over intranet by using OBS Browser+ on a Windows ECS

1. Start
2. Log in to a Windows ECS.
3. Check whether the intranet DNS is configured on the Windows ECS.
   - No: Configure the intranet DNS.
   - Yes: Download OBS Browser+.
4. Download OBS Browser+.
5. Log in to OBS Browser+.
6. Use OBS Browser+.
7. End

**Procedure**

**Step 1** Log in to a Windows ECS.
1. Log in to **HUAWEI CLOUD** and click **Console**.
2. On the homepage of the console, choose **Computing > Elastic Cloud Server**.
3. Select an ECS and log in to the ECS.

   A Windows ECS provides two login modes, VNC remote login and MSTSC. For details, see **Purchasing and Logging In to a Windows ECS**.

**Step 2 Check whether the intranet DNS is configured on the Windows ECS.**

On the Windows ECS, you can view the current DNS configuration by using the graphical user interface (GUI) or command line interface (CLI). This section uses the CLI as an example to explain how to view the DNS configuration.

1. After logging in to the ECS, open the CLI.
2. Run the `ipconfig /all` command to check whether the DNS server is at the intranet DNS address in the region where the ECS resides.

   ![NOTE](image)

   HUAWEI CLOUD provides different intranet DNS server addresses for different regions. For details, see **What Are the Private DNS Server Addresses Provided by the DNS Service?**

   - If not, go to **Step 3**.
   - If yes, go to **Step 4**.

**Step 3 Configure the intranet DNS.**

Change the DNS server address of the ECS to the intranet DNS provided by HUAWEI CLOUD. You can change the DNS address of the VPC subnet or modify the local DNS configuration to achieve this.

- **Methods 1: Changing the DNS server address of the VPC subnet**

  Locate the VPC where the ECS resides and change the DNS server address of the VPC subnet to the intranet DNS address. In this manner, ECSs in the VPC can use the intranet DNS for resolution and thereby you can access OBS on HUAWEI CLOUD intranet. For details, see **Modifying a Subnet**.

   ![NOTE](image)

   The intranet DNS server address must be selected based on the region where the ECS resides. For details, see **What Are the Private DNS Server Addresses Provided by the DNS Service?**

- **Method 2: Modifying the local DNS configuration**

  The intranet DNS configured using this method becomes invalid once the ECS is restarted. Therefore, you need to reconfigure the intranet DNS after each restart of the ECS. This example uses the CLI tool to modify the DNS configuration locally.

1. Open the cmd window.
2. Run the following command to configure the IP address of the primary DNS server:

   ```
   netsh interface ip set dns name="Local connection" source=static address=Intranet DNS server register=primary
   ```
NOTE

- **Local connection**: NIC name. Use the actual NIC name when configuring the local DNS.
- **Intranet DNS server address**: Select the intranet DNS server address based on the region where the ECS resides. For details, see *What Are the Private DNS Server Addresses Provided by the DNS Service?*

3. (Optional) Run the following command to configure the IP address of the backup DNS server:

```bash
netsh interface ip add dns name="Local connection" addr=Alternative DNS server address index=2
```

NOTE

- **Local connection**: NIC name. Use the actual NIC name when configuring the local DNS.
- **Alternative DNS server address**: The DNS server used when the primary DNS server is faulty, unavailable, or cannot resolve the requested domain name. Therefore, you can set this parameter to the IP address of the HUAWEI CLOUD intranet DNS server. (You need to select the intranet DNS server address based on the region where the ECS resides. For details, see *What Are the Private DNS Server Addresses Provided by the DNS Service?*) You can also set this parameter to the IP address of a public DNS server.

**Step 4** Download OBS Browser+.

For details, see *Downloading OBS Browser+*.

**Step 5** Log in to OBS Browser+.

OBS Browser+ uses accesses OBS over the internet by default. Therefore, when you log in to OBS Browser+ to add an account, set **Service** and **Server Address** as follows:

- **Service**: Select *Other object storage services*.
- **Server Address**: Enter the OBS domain name for the region where your ECS resides and the port number. The HTTPS port number is 443 and the HTTP port number is 80. An HTTPS server is used by default.

Example: `obs.ap-southeast-1.myhuaweicloud.com:443`

NOTE

For details about OBS regions and endpoints, see *Regions and Endpoints*.

**Step 6** Start to use OBS Browser+.

After logging in to OBS Browser+, you can access OBS over HUAWEI CLOUD intranet from the Windows ECS to perform basic data access operations and configure other advanced settings.

For details, see the *Introduction to OBS Browser+*.

----End
4.3 Accessing OBS over Intranet by Using obsutil on a Linux ECS

obsutil is a command line tool applicable to Windows and Mac operating systems. You can configure the intranet DNS server address to access OBS over intranet on a HUAWEI CLOUD Linux ECS. The process and procedure are described as follows.

**Process**

**Figure 4-3** The process of accessing OBS over intranet by using OBS Browser on a Linux ECS

![Diagram showing the process of accessing OBS over intranet by using obsutil on a Linux ECS]

**Procedure**

**Step 1** Log in to the Linux ECS.

1. Log in to **HUAWEI CLOUD** and click **Console**.
2. On the home page of the console, choose **Computing > Elastic Cloud Server**.
3. Select an ECS and log in to the ECS.
   The login mode varies according to the login authentication mode set during the Linux ECS purchase. For details about how to log in to the ECS, see **Purchasing and Logging In to a Linux ECS**.

**Step 2** Check whether the intranet DNS is configured on the Linux ECS.
1. Log in to the Linux ECS and open the CLI.
2. Run the `cat /etc/resolv.conf` command to check whether the IP address after `nameserver` in the first line is the intranet DNS address of the region where the current ECS resides.

   **NOTE**
   HUAWEI CLOUD provides different intranet DNS server addresses for different regions. For details, see **What Are the Private DNS Server Addresses Provided by the DNS Service?**
   - If no, go to **Step 3**.
   - If yes, go to **Step 4**.

**Step 3** Configure the Intranet DNS.
Change the DNS server address of the ECS to the intranet DNS provided by HUAWEI CLOUD. You can change the DNS address of the VPC subnet or modify the local DNS configuration to achieve this.

- **Methods 1: Changing the DNS server address of the VPC subnet**
  Locate the VPC where the ECS resides and change the DNS server address of the VPC subnet to the intranet DNS address. In this manner, ECSs in the VPC can use the intranet DNS for resolution and thereby you can access OBS on HUAWEI CLOUD intranet. For details, see **Modifying a Subnet**.

   **NOTE**
   The intranet DNS server address must be selected based on the region where the ECS resides. For details, see **What Are the Private DNS Server Addresses Provided by the DNS Service?**

- **Method 2: Modifying the local DNS configuration**
  The following uses an ECS running 64-bit CentOS 6.x as an example to describe how to modify the local DNS configuration.
  a. Open the CLI.
  b. Run the following command to open the `/etc/resolv.conf` file:
     ```bash
     vi /etc/resolv.conf
     ```
  c. Press i to enter the editing mode. In the `/etc/resolv.conf` file, add the intranet DNS server address before the existing DNS server address in the following format:
     ```bash
     nameserver Intranet DNS server address
     ```
Intranet DNS server address: Select the intranet DNS server address based on the region where the ECS resides. For details, see What Are the Private DNS Server Addresses Provided by the DNS Service?

The IP address of the new DNS server must come before all existing DNS IP addresses.

DNS servers are selected in the sequence of nameserver. A new DNS server is selected only when the previous DNS server is faulty, unavailable, or cannot resolve the requested domain name. Therefore, if you want to switch to the public network access mode, you need to change the first line of the DNS address to a public DNS server address or add a public DNS server address before the existing DNS server address.

d. Press ESC and enter :wq! to save the settings and close the file.

NOTE

The modified DNS server address takes effect immediately after you save the modification to the /etc/resolv.conf file.

Step 4 Download obsutil.

For details about the latest version of obsutil and download link, see Download and Installation.

Step 5 Configure obsutil.

Before using, you need to configure the interconnection between obsutil and OBS. Parameters include OBS endpoints and access keys (AK and SK). For details, see Performing Initial Configuration in the obsutil tool guide.

NOTE

The OBS endpoint needs to be entered according to the region where the ECS resides. For details about OBS regions and endpoints, see Regions and Endpoints.

Step 6 Use obsutil.

After obsutil is successfully configured, you can access OBS over HUAWEI CLOUD intranet on the Linux ECS to perform basic data access operations and other advanced settings.

For details, see the following topics:

- Uploading an Object
- Downloading an Object

For details about operations, see Introduction to obsutil.

----End
5 Using a User-Defined Domain Name to Host a Static Website

5.1 Overview

OBS allows you to access static websites hosted by OBS using user-defined domain names. This section uses a typical scenario as an example to describe how to use a user-defined domain name to configure static website hosting. Before starting the configuration, you may need to learn more about static website hosting.

Scenario Introduction

Company A has a large number of files to archive but it does not want to put efforts on storage resources. Therefore, the company subscribes to OBS for hosting static websites and expects that the user names owned by the company can access the static resources through a user-defined domain name. See Figure 5-1.

Figure 5-1 Using a user-defined domain name to access hosted static website
Data Planning

Table 5-1 describes the data to be planned before this configuration.

<table>
<thead>
<tr>
<th>Item</th>
<th>Description</th>
<th>Example</th>
</tr>
</thead>
<tbody>
<tr>
<td>User-defined domain name</td>
<td>User’s own domain name</td>
<td><a href="http://www.example.com">www.example.com</a></td>
</tr>
<tr>
<td>Static website homepage</td>
<td>Indicates the index page that is returned when you access a static website, that is, the homepage.</td>
<td>index.html</td>
</tr>
<tr>
<td>Default 404 Page</td>
<td>When an incorrect static website path is accessed, the 404 error page is returned.</td>
<td>error.html</td>
</tr>
</tbody>
</table>

- The contents of index.html are as follows:
  ```html
  <html>
  <head>
    <title>Hello OBS!</title>
    <meta charset="utf-8">
  </head>
  <body>
    Welcome to use OBS static website hosting.
    This is the homepage.
  </body>
  </html>
  ```

- The contents of error.html are as follows:
  ```html
  <html>
  <head>
    <title>Hello OBS!</title>
    <meta charset="utf-8">
  </head>
  <body>
    Welcome to use OBS static website hosting.
    This is the 404 error page.
  </body>
  </html>
  ```

5.2 Static Website Hosting

Process of Static Website Hosting

You need to create a bucket on OBS Console to store static website resources, enable static website hosting for the bucket, and bind the user-defined domain name to the newly created bucket using the user-defined domain name binding function provided by OBS. Then, create and configure domain name hosting using Domain Name Service (DNS) so that a user-defined domain name can be used to access the static website hosted on the OBS. Specific operations are as follows:

1. **Register a domain name.**
2. Create a bucket.
3. Upload static website files.
4. Host the static website on OBS.
5. Bind a user-defined domain name.
6. Create and configure domain name hosting.
7. Verify the configuration.

Procedure

Step 1 Register a domain name.
If you have a registered domain name, skip this step.
If you do not have a registered domain name, register one with a registrar. In this scenario, the example domain name www.example.com is used. In practice, you need to replace the domain name with the one you actually planned.

Step 2 Create a bucket.
There is no special requirement for bucket names. You only need to create a bucket for storing static website files as prompted. The following example describes how to create a bucket named example:

1. Open OBS Console and log in to the console as prompted.
2. Click Create Bucket in the upper part of the page.
3. Set the following parameters in the dialog box that is displayed:
   - Region: Select a region close to the service according to the proximity principle.
   - Storage Class: It is recommended that you select Standard.
   - Bucket Name: Enter example.
   - Bucket Policies: Select Public Read to allow any user to access objects in the bucket.
4. Click Create Now. The bucket is created.

Step 3 Upload static website files.
Prepare the static website files to be uploaded and repeat the following steps on OBS Console until all static website files are uploaded to bucket created in Step 2.

**NOTE**
In regions where batch uploading is available, you can upload a maximum of 100 files in a batch, with a total size not more than 5 GB. If there are a large number of website files, you are advised to use OBS Browser+ to upload them. For details, see Using OBS Browser+ to Upload a File or Folder.

1. Click the name of the target bucket to go to the bucket overview page, and then click Object in the navigation pane on the left.
2. Click **Upload Object**. A dialog box is displayed. For details, see **Figure 5-2**.

**Figure 5-2 Uploading an object**

3. Add the files to be uploaded.

**NOTE**

- The static website files cannot be encrypted for upload.
- It is recommended that you select **Standard** for the storage class. If the storage class of static website files is **Archive**, you need to restore the files first before accessing it. For details, see **Restoring an Archive File on OBS**.
- The website home page file (index.html) and 404 error page (error.html) must be stored in the root directory of the bucket.

4. Click **Upload** to upload the files.

**Step 4 Configure static website hosting.**

After uploading the static website files, you need to perform the following steps to set the bucket to the static website hosting mode.

**NOTE**

You can also redirect the entire static website to another bucket or domain name. For details, see **Configuring Redirection**.

1. Click the bucket that you want to configure. On the **Overview** page that is displayed, choose **Basic Configurations > Static Website Hosting** on the navigation pane on the left.

2. Click **Configure Static Website Hosting**.

3. In the dialog box that is displayed, select **Use this bucket to host a website**, set **Default Home Page** to index.html in the data plan, and set **Default 404 Error Page** to error.html in the data plan. See **Figure 5-3**.
You can also configure redirection rules based on service requirements to implement website content redirection. For details, see Configuring Static Website Hosting.

4. Click OK.

Step 5 Bind a user-defined domain name.

To bind a user-defined domain name to OBS, perform the following steps:

1. Click the bucket name to go to the Overview page. In the navigation tree on the left, choose Domain Name Management.
2. Click Bind User Domain Name, and enter www.example.com in the User Domain Name text box, as shown in Figure 5-4.

Figure 5-4 Binding a user domain name

3. Click OK. The user-defined domain name is bound to the bucket domain name.
Step 6  Create and configure domain name hosting.

To facilitate unified management of your user-defined domain names and static websites and implement cloud-based services, you can directly manage your user-defined domain names on HUAWEI CLOUD DNS. After the hosting is configured, you can perform subsequent management of the domain name on DNS, including managing record sets and PTR records, as well as creating wildcard DNS records.

**NOTE**

Alternatively, you can add a CNAME record to the DNS at the DNS registrar, mapping to the website domain name hosted by the bucket. For example: If the region of bucket www.example.com is AP-Hong Kong, you need to add a CNAME record whose value is www.example.com CNAME www.example.com.obs-website.ap-southeast-1.myhuaweicloud.com at your DNS registrar.

To create and configure domain name hosting on DNS, perform the following steps:

1. Add a public zone.
   Use the root domain name example.com created in Step 1 as the name of the public zone to be created. For details, see "Create a Public Zone" in Configuring Record Sets for a Website.

2. Add a CNAME record.
   In DNS, add a record set for the sub-domain name www.example.com of the hosted domain name, to map the CNAME of the sub-domain name to the static website domain name hosted by OBS. Configure the parameters as follows:
   - **Name**: Enter www.
   - **Type**: Select CNAME-Canonical name.
   - **Line**: Select Default.
   - **TTL (s)**: Retain the default value.
   - **Value**: Domain name mapped to the CNAME. If CDN acceleration is disabled when a user-defined domain name is bound, enter the bucket access domain name. If CDN acceleration is enabled, set this parameter to the acceleration domain name (CNAME) provided by CDN.

   For details, see Adding a CNAME Record Set.

3. Change the DNS server address at your domain name registrar.
   At your domain name registrar, change the DNS server address in the NS record of the root domain name to the cloud DNS server address. The specific address is the NS value of the public zone in DNS.

   For details about how to change the IP address of the DNS server, see section "Change the DNS Servers of the Domain Name" in Configuring Record Sets for a Website.

   **NOTE**

   Generally, the update takes effect within 48 hours, but the time may vary depending on domain name registrars.

Step 7  Verify that the configuration is successful.

- Enter www.example.com in the address bar of the browser to check whether the default homepage can be accessed. For details, see Figure 5-5.
In the web browser, enter a static file access address that does not exist in a bucket. For example: www.example.com/imgs to verify that the 404 error page can be returned. See Figure 5-6.

Figure 5-6 Default 404 Page

Welcome to use OBS static website hosting.
This is the 404 error page.

---End

5.3 Updating a Static Website

If you need to update a static file (such as a picture, music file, HTML file, or CSS file) on a website, you can upload the static file again. Note that the newly uploaded files in the same path of OBS overwrite the existing files with the same names by default. To avoid file overwriting, you can enable the versioning function of OBS. With versioning enabled, OBS can store multiple versions of a static file. You can quickly search for and restore different versions or restore data in the event of mis-operations or application faults.
Enabling Versioning

**Step 1** Log in to OBS Console.

**Step 2** In the bucket list, click the target bucket to go to the **Overview** page.

**Step 3** In the **Basic Information** area, locate **Versioning** and click **Edit** to its right. See **Figure 5-7**.

**Figure 5-7 Versioning**

- Enable
- Suspend

Suspend the creation of object versions for any operation but retain any existing object versions.

**Step 4** Select **Enable** and click **OK** to enable versioning for objects in the bucket.

----End

For more information about versioning, see **Versioning Overview**.

Updating Static Files

**Step 1** Log in to OBS Console.

**Step 2** In the bucket list, click the target bucket to go to the **Overview** page.

**Step 3** In the navigation tree on the left, click **Object**.

**Step 4** Click **Upload Object**, or select the folder where the file to be updated is located and click **Upload Object**. A dialog box is displayed. See **Figure 5-8**.
**Figure 5-8** Uploading an object

**Step 5** Add the files to be uploaded.

- **NOTE**
  - The static website files cannot be encrypted for upload.
  - It is recommended that you select **Standard** for the storage class. If the storage class of static website files is **Archive**, you need to restore the files first before accessing it. For details, see Restoring an Archive File on OBS.

**Step 6** Click **Upload** to complete the upload.

The newly uploaded file with the same name in the same path is displayed as the latest version in the object list. Each time, only the latest version of the file is accessed. In this way, the static website file can be updated.

----End
6 Enterprise Data Access Control

6.1 Introduction to OBS Access Control

By default, only the resource owner can access OBS resources. Without authorization, other users cannot access OBS. OBS offers multiple methods to help you to assign the resource permissions to others. Resource owners can formulate different access control schemes based on service requirements to ensure data security.

OBS Access Control Mechanisms

OBS provides multiple permission control mechanisms, including IAM permissions, IAM agencies, object access restriction, object ACLs, bucket ACLs, and bucket policies. Table 6-1 describes the mechanisms and application scenarios.

Table 6-1 OBS access control mechanisms

<table>
<thead>
<tr>
<th>Mechanism</th>
<th>Description</th>
<th>Application Scenario</th>
</tr>
</thead>
</table>
| IAM permissions | IAM permissions define the actions that can be performed on your cloud resources. In other words, IAM permissions specify what actions are allowed or denied. After an IAM user is created, the administrator needs to add the user to a group. IAM can grant the user group required OBS access permissions, and then all users in the group automatically inherit the permissions of the user group. For details about OBS permissions that can be authorized by IAM, see Permissions Management. | • Controlling permissions to cloud resources as a whole  
• Controlling permissions to all OBS buckets and objects |
<table>
<thead>
<tr>
<th>Mechanism</th>
<th>Description</th>
<th>Application Scenario</th>
</tr>
</thead>
<tbody>
<tr>
<td>IAM agency</td>
<td>Delegates other accounts or services to access OBS. A delegated party can manage OBS resources on behalf of the delegating party. This achieves secure and efficient service management.</td>
<td>Reading data in private OBS buckets using other cloud services, for example, OBS private bucket retrieval</td>
</tr>
</tbody>
</table>
| Time-limited access to objects | Provides a URL that is accessible within a specified period of time. In this way, anonymous users can download software from objects only within the specified period. For details, see Sharing a Folder. | ● Granting a third party the temporary permission to download objects without user authentication  
● Granting permissions to other HUAWEI CLOUD accounts or users for only temporary access |
| Object ACL             | Controls access to objects for accounts or user groups. Object owners can configure the object access control list (ACL) to grant basic read and write permissions to specified accounts or user groups.  
**NOTE**  
- By default, an object ACL is created when the object is uploaded. The object owner has full control over the object.  
- The owner of an object is the account that uploads the object, who may not be the owner of the bucket to which the object belongs. For example, account B is granted the permission to access a bucket of account A, and account B uploads a file to the bucket. In that case, instead of the bucket owner account A, account B is the owner of the object.  
|                                                                       | ● Object-level access control is required. A bucket policy can control access permissions for an object or a set of objects. If you want to further specify an access permission for an object in the set of objects for which a bucket policy has been configured, then the object ACL is recommended for easier access control over single objects.  
● An object is accessed through a URL. Generally, if you want to grant anonymous users the permission to read an object through a URL, use the object ACL. |
<table>
<thead>
<tr>
<th>Mechanism</th>
<th>Description</th>
<th>Application Scenario</th>
</tr>
</thead>
</table>
| **Bucket ACL** | Controls access to buckets for accounts or user groups. Bucket owners can configure the bucket ACL to grant basic read and write permissions to specified accounts or user groups. | • Grant the log delivery user with the write access to the target bucket, so that access logs can be delivered to the target bucket.  
• Grant an account with the read and write access to a bucket, so that data in the bucket can be shared or the bucket can be mounted. For example, if account A grants the bucket read and write permissions to account B, then account B can access the bucket by using the API and SDK, and can add an external bucket through OBS Browser+. |
| **NOTE** | • By default, a bucket ACL is created upon the creation of the bucket. The bucket owner has full control over the bucket.  
• Bucket ACLs do not provide fine-grained permission control. Generally, IAM permissions and bucket policies are recommended for permission access control. | |
| **Bucket policy** | Bucket policies provide centralized access control on OBS resources, and define which operations on which cloud resources are allowed. They are the extension and supplement of ACLs of buckets and objects. | • If no IAM permissions is used for access permission control and you want to authorize other accounts the permission to access your OBS resources, you can use bucket policies to authorize such permissions.  
• If you want to authorize IAM users different access permissions to different buckets, you can configure different bucket policies for buckets.  
• If you want to authorize other accounts the permission to access your buckets, you can use bucket policies to authorize such permissions. |
Object policy

An object policy is also a part of a bucket policy. A bucket policy can be applied to multiple or all objects in a bucket. An object policy applies to specified objects. Only actions and conditions related to the object can be configured in an object policy.

<table>
<thead>
<tr>
<th>Mechanism</th>
<th>Description</th>
<th>Application Scenario</th>
</tr>
</thead>
<tbody>
<tr>
<td>Object policy</td>
<td></td>
<td>Configuring for a single object. For example, you can directly configure a policy for the object after an object is uploaded.</td>
</tr>
</tbody>
</table>

For details about access control mechanisms when grantees and authorized resources are involved, see Table 6-2.

Table 6-2 OBS access control mechanisms

<table>
<thead>
<tr>
<th>Mechanism</th>
<th>Grantee</th>
<th>Authorized Resource</th>
<th>Granted Operation</th>
<th>Condition Configuration</th>
</tr>
</thead>
<tbody>
<tr>
<td>IAM permissions</td>
<td>IAM users</td>
<td>All OBS resources except specified OBS resources or resource sets</td>
<td>All permissions to access OBS</td>
<td>Not supported</td>
</tr>
<tr>
<td>IAM agency</td>
<td>● Account services</td>
<td>All OBS resources except specified OBS resources or resource sets</td>
<td>All permissions to access OBS</td>
<td>Time limitation configuration (permanent or one-day)</td>
</tr>
<tr>
<td>Time-limited access to objects</td>
<td>Anonymous users</td>
<td>Objects</td>
<td>Obtains the content and metadata of an object.</td>
<td>Time limitation configuration</td>
</tr>
<tr>
<td>Mechanism</td>
<td>Grantee</td>
<td>Authorized Resource</td>
<td>Granted Operation</td>
<td>Condition Configuration</td>
</tr>
<tr>
<td>-----------</td>
<td>---------</td>
<td>---------------------</td>
<td>-------------------</td>
<td>-------------------------</td>
</tr>
</tbody>
</table>
| Object ACL | • Account s  
• Anonymous users  
• Registered user groups | Objects | • Obtains the content and metadata of an object.  
• Obtains the content and metadata of a specified object.  
• Obtains the ACL for an object.  
• Obtains the ACL for an object of a specified version.  
• Configures object ACL.  
• Configures the ACL for an object of a specified version. | Not supported |
| Bucket ACL | • Account s  
• Anonymous users  
• Registered user groups  
• Log delivery user groups | Buckets | • Identifies whether a bucket exists.  
• Lists objects in a bucket, and obtains the bucket metadata.  
• Lists multi-version objects in a bucket.  
• Lists multipart upload tasks.  
• Performs PUT upload, POST upload, multipart upload, initialization of uploaded parts, and merging of parts.  
• Deletes an object.  
• Deletes an object of a specified version.  
• Obtains the ACL for a bucket.  
• Configures the ACL for a bucket. | Not supported |
### OBS Access Control Principles

- **Least privilege**
  
  Only the minimum permissions required for executing tasks are granted to IAM users or accounts. For example, if an IAM user only needs to upload and download objects to a specified directory, you do not need to assign the user the read and write permissions to the bucket.

- **Separation of duties**
  
  You are advised to assign different IAM users under an account to manage different OBS resources and permissions. For example, IAM user A is responsible for assigning permissions, and other IAM users managing OBS resources.

- **Restricted conditions**
  
  Configure refined conditions for bucket policies to restrict scenarios where a bucket policy takes effect, in order to enhance the security of resources in a bucket. For example, OBS is configured to accept only access requests from a specific IP address.

### How Does Authorization Work When Multiple Access Control Mechanisms Co-Exist?

Based on the least-privilege principle, decisions default to deny, and an explicit deny statement always take precedence over an allow statement. For example, IAM permissions grant a user the access to an object, a bucket policy denies the user's access to that object, and there is no ACL. Then access will be denied.

If no method specifies an allow statement, then the request will be denied by default. Only if no method specifies a deny statement and one or more methods...
specify an allow statement, will the request be allowed. For example, if a bucket has multiple bucket policies with allow statements, the adding of a new bucket policy with an allow statement will simply add the allowed permissions to the bucket, but the adding of a new bucket policy with a deny statement will result in a re-arrangement of the permissions. The deny statement will take precedence over allowed statements, even the denied permissions are allowed in other bucket policies.

**Figure 6-1** Authorization process

![Authorization process diagram](image)

**Figure 6-2** is a matrix of the IAM permissions, bucket policies, and ACLs (allow and deny effects).

**Figure 6-2** Matrix of the IAM permissions, bucket policies, and ACLs (allow and deny effects)

<table>
<thead>
<tr>
<th>Bucket Policy</th>
<th>IAM Policy</th>
<th>ACL</th>
</tr>
</thead>
<tbody>
<tr>
<td>Deny</td>
<td>Deny</td>
<td>Allow</td>
</tr>
<tr>
<td>Deny</td>
<td>Deny</td>
<td>Default Deny</td>
</tr>
<tr>
<td>Allow</td>
<td>Deny</td>
<td>Allow</td>
</tr>
<tr>
<td>Allow</td>
<td>Deny</td>
<td>Default Deny</td>
</tr>
<tr>
<td>Default Deny</td>
<td>Allow</td>
<td>Deny</td>
</tr>
<tr>
<td>Default Deny</td>
<td>Deny</td>
<td>Deny</td>
</tr>
</tbody>
</table>

**Related Concepts**

- **Account**: An account is automatically created when a user registers with HUAWEI CLOUD. This account has full access permissions for the resources and IAM users under the account.
- **Administrator**: A user who has the `admin` permission created in IAM and manages IAM users on behalf of the account to ensure security of an account and resources.
NOTE

admin is a user group preset on the IAM system and has all operation permissions. An administrator added to the admin user group has the same resource management and user management permissions as the account.

- IAM user: A user created by the administrator in IAM. An IAM user uses cloud services and corresponds to an employee, system, or application. IAM users have identity credentials (passwords and access keys) and can log in to the management console or access APIs.

6.2 Access Management on Department Public Data

An enterprise has a large number of files to archive but it does not want to put efforts on storage resources. Therefore, this enterprise subscribes to OBS for storing the files, and expects that staff in different departments have different access permissions. By doing so, data access permissions of staff in different departments are isolated.

The enterprise expects that administrators have the full control permission to department public data stored on OBS, and that common users can only read those data. Figure 6-3 shows the logical relationships.

Figure 6-3 Logical relationship

Solution and Process

In this scenario, you can assign permissions by configuring IAM permissions. Set the permission of the user group containing common users to Tenant Guest, so that common users can access OBS as guests and have only the read permission. Figure 6-4 shows the process.
Figure 6-4 Flowchart of managing access to department public data

Procedure

Step 1 Create an administrator.
1. Log in to the HUAWEI CLOUD console using the enterprise account.
2. On the console homepage, choose Service List > Management & Deployment > Identity and Access Management to access the IAM console.
3. On the IAM console, choose User in the left navigation tree.
4. On the User page, click Create User. On the page that is displayed, enter a username and configure the following parameters:
   - Select Password for Credential Type.
   - Select admin from the drop-down list of User Groups.
5. Click Next. Select Set manually for Password Type.
6. Enter the email address, mobile number, password, and confirm password.
7. Click OK.

Step 2 Create a user group with the read-only permission.
1. On the IAM console, choose User Group in the left navigation tree.
2. Click Create User Group, and enter a user group name and description.
3. Click OK.
   The user group list is displayed, including the newly created user group.
4. Locate the newly created user group, and click Configure Permission in the Operation column.
5. In the User Group Permissions area on the displayed page, select OBS and click Configure Policy.
6. In the available policy list, select the Tenant Guest policy.
7. Click OK to save the permission for the user group.

**Step 3 Create a common user.**

1. On the IAM console, choose User in the left navigation tree.
2. On the User page, click Create User. On the page that is displayed, enter a username and configure the following parameters:
   - Select Password for Credential Type.
   - Select the user group created in Step 2 for User Groups.
3. Click Next. Select Set manually for Password Type.
4. Enter the email address, mobile number, password, and confirm password.
5. Click OK.

**Step 4 Verify the user permission.**

After the permission is granted, you can verify the permissions using OBS Console, OBS Browser+, APIs, and SDKs. This section takes OBS Console as an example to present how to verify the read-only permission of common users on department public data.

1. Log in to OBS Console as a common user and check whether you have the permission to access the OBS page.
   - If a message indicating that you do not have the permission to access the page is displayed, you cannot read data in the bucket. In this case, check whether the user permission is correctly configured.
   - If a bucket list is displayed, you have the permission to read the bucket list. Go to the next step.
2. Click the bucket to be operated. On the Summary page that is displayed, click Objects to view the list of objects.
   - If the data cannot be obtained and the message Access denied is displayed, you have no permission to read data in the bucket. In this case, check whether the user permission is correctly configured.
   - If the data is displayed, you have the read permission. Go to the next step.
3. On the Objects page, perform operations including uploading and deleting objects.
   - If the write and delete operations can be performed, it indicates the read-only permission fails to be granted. Check whether the user permission configuration is correct.
   - If not, the read-only permission for common users is correctly configured.

---

### 6.3 Data Sharing Among Departments/Projects

An enterprise has data that needs to be shared among different departments or projects. To reduce the risks of mistaken deletion and tampering of shared data,
the data can only be downloaded but not modified or deleted by users of other departments.

In this scenario, department A shares data in the bucket **example-bucket** to department B, allowing users of department B to download the data. This case describes how to leverage the least privilege principle to control access permissions for the shared data. **Figure 6-5** shows the logical relationships among administrators, users, and buckets for data sharing between the two departments in this scenario.

**Figure 6-5 Logical relationship**

![Logical relationship diagram](image)

**Solution and Process**

In this scenario, the administrator of department A can use bucket policies to implement permission control, so that users of department B can only download but not modify or delete the shared data. **Figure 6-6** illustrates the bucket policy configuration process.
**Figure 6-6** Configuration process of permission control for data sharing

**Prerequisites**

Administrators and common users of departments A and B have been created on IAM. For details about how to create an IAM user, see [Creating an IAM User](#).

**NOTE**

The administrator of department A needs to perform operations such as creating buckets and configuring bucket policies. Therefore, when creating an administrator, the user group to which the administrator belongs must be granted at least the **Tenant Administrator** permissions of OBS.

**Procedure**

**Step 1  Create a bucket.**

1. Log in to the HUAWEI CLOUD console as the administrator of department A.
2. On the console homepage, choose **Service List > Storage > Object Storage Service** to access OBS Console.
3. On OBS Console, click **Create Bucket** in the upper right corner.

4. Select a region, storage class, and bucket policy, and then enter the bucket name.

   **NOTE**

   To ensure data security, set **Bucket Policy** to **Private** and set other parameters as prompted.

5. Click **Create Now**. The bucket is created.

**Step 2 Authorize users with the upload permission.**

If the OBS policy for the user group to which users of department A belong is **Tenant Administrator**, skip this step and go to **Step 3**. If the OBS policy is not configured or the policy is set to **OBS Buckets Viewer**, **OBS ReadOnlyAccess**, or **Tenant Guest**, the administrator of department A needs to perform the following steps to grant the upload permission to users of department A.

1. On OBS Console, click the name of the bucket where the shared data is stored and the **Summary** page of the bucket is displayed.

2. In the left navigation pane, choose **Permissions**. On the page that is displayed, click the **Bucket Policies** tab.

3. Click **Create Bucket Policy** under **Custom Bucket Policies**.

4. Create a custom bucket policy by referring to **Table 6-3**.

   **Table 6-3 Parameters for creating a policy with the upload permission**

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Configuration</th>
</tr>
</thead>
<tbody>
<tr>
<td>Policy Mode</td>
<td>Select <strong>Read-only</strong>.</td>
</tr>
<tr>
<td>Principal</td>
<td>Select <strong>Include</strong> and <strong>Current account</strong>, and then select an authorized user from the drop-down list. Select users of department A, to whom you want to authorize the upload permission.</td>
</tr>
<tr>
<td>Resources</td>
<td>Select <strong>Include</strong> and enter the resource name based on the scope of the shared data.</td>
</tr>
<tr>
<td></td>
<td>- If all objects in the bucket need to be shared, enter <code>*</code>.</td>
</tr>
<tr>
<td></td>
<td>- If you need to share only a folder or a set of objects in the bucket, enter the name of the folder (for example, <code>example-folder/</code>) or an object set with a wildcard character (for example, <code>*.doc</code>, indicating all objects whose name ends with <code>.doc</code>). You can also enter multiple resource names, separating them using commas (,).</td>
</tr>
</tbody>
</table>

5. Click **OK**. If a message similar to **Bucket policy created successfully** is displayed, the upload permission is successfully authorized to the selected users of department A.

**Step 3 Authorize users with the download permission.**
If the policy for the user group to which users of department B belong is **Tenant Administrator** or **Tenant Guest**, skip this step and go to step 2. If the OBS policy is not configured or the policy is set to **OBS Buckets Viewer** or **OBS ReadOnlyAccess**, the administrator of department A needs to perform the following steps to grant the download permission to users of department B.

1. On OBS Console, click the name of the bucket where the shared data is stored and the *Overview* page of the bucket is displayed.
2. In the left navigation pane, choose *Permissions*. On the page that is displayed, click the *Bucket Policies* tab.
3. Click **Create Bucket Policy** under **Custom Bucket Policies**.
4. Create a custom bucket policy by referring to **Table 6-4**.

**Table 6-4** Parameters for creating a policy that grants the download permission to users

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Configuration</th>
</tr>
</thead>
<tbody>
<tr>
<td>Policy Mode</td>
<td>Select <strong>Customized</strong>.</td>
</tr>
<tr>
<td>Effect</td>
<td>Select <strong>Allow</strong>.</td>
</tr>
<tr>
<td>Principal</td>
<td>Select <strong>Include</strong> and <strong>Current account</strong>, and then select a user from the drop-down list. Select users of department B, to whom you want to authorize the download permission.</td>
</tr>
<tr>
<td>Resources</td>
<td>Select <strong>Include</strong> and enter the resource name based on the scope of the shared data.</td>
</tr>
<tr>
<td></td>
<td>- If all objects in the current bucket need to be shared, leave the resource name blank.</td>
</tr>
<tr>
<td></td>
<td>- If you need to share only a folder or a set of objects in the bucket, enter the name of the folder (for example, <code>example-folder/</code>) or an object set with a wildcard character (for example, <code>*.doc</code>, indicating all objects whose name ends with <code>.doc</code>). You can also enter multiple resource names, separating them using commas (,).</td>
</tr>
<tr>
<td>Actions</td>
<td>Select <strong>Include</strong>, and select <strong>Get</strong> and <strong>List</strong> under <strong>General</strong> from the drop-down list of <strong>Action Name</strong>.</td>
</tr>
</tbody>
</table>

5. Click **OK**. If a message similar to **Bucket policy created successfully** is displayed, the download permission is successfully authorized to the selected users of department B.

**Step 4** Prevent users from modifying and deleting the shared data.

1. On OBS Console, click the name of the bucket where the shared data is stored and the *Overview* page of the bucket is displayed.
2. In the left navigation pane, choose *Permissions*. On the page that is displayed, click the *Bucket Policies* tab.
3. Click **Create Bucket Policy** under **Custom Bucket Policies**.
4. Create a custom bucket policy by referring to Table 6-5.

Table 6-5 Parameters for creating a policy to prevent write and deletion

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Configuration</th>
</tr>
</thead>
<tbody>
<tr>
<td>Policy Mode</td>
<td>Select <strong>Customized</strong>.</td>
</tr>
<tr>
<td>Effect</td>
<td>Select <strong>Deny</strong>.</td>
</tr>
<tr>
<td>Principal</td>
<td>Select <strong>Include</strong> and <strong>Current account</strong>, and then select an authorized user</td>
</tr>
<tr>
<td></td>
<td>from the drop-down list. Select users of department B, whom you want to prevent</td>
</tr>
<tr>
<td></td>
<td>from writing and deleting the shared data.</td>
</tr>
<tr>
<td>Resources</td>
<td>Select <strong>Include</strong> and enter the resource name based on the scope of the</td>
</tr>
<tr>
<td></td>
<td>shared data.</td>
</tr>
<tr>
<td></td>
<td>– If all objects in the bucket need to be shared, enter *.</td>
</tr>
<tr>
<td></td>
<td>– If you need to share only a folder or a set of objects in the bucket,</td>
</tr>
<tr>
<td></td>
<td>enter the name of the folder (for example, <code>example-folder/</code>) or an object</td>
</tr>
<tr>
<td></td>
<td>set with a wildcard character (for example, <code>*.doc</code>, indicating all objects</td>
</tr>
<tr>
<td></td>
<td>whose name ends with <code>.doc</code>). You can also enter multiple resource names,</td>
</tr>
<tr>
<td></td>
<td>separating them using commas (,).</td>
</tr>
<tr>
<td>Actions</td>
<td>Select <strong>Include</strong> and select the following six actions under <strong>Object</strong> from</td>
</tr>
<tr>
<td></td>
<td>the drop-down list of <strong>Action Name</strong>.</td>
</tr>
<tr>
<td></td>
<td>– PutObject</td>
</tr>
<tr>
<td></td>
<td>– PutObjectAcl</td>
</tr>
<tr>
<td></td>
<td>– PutObjectVersionAcl</td>
</tr>
<tr>
<td></td>
<td>– DeleteObject</td>
</tr>
<tr>
<td></td>
<td>– DeleteObjectVersion</td>
</tr>
<tr>
<td></td>
<td>– AbortMultipartUpload</td>
</tr>
</tbody>
</table>

5. Click **OK**. If a message similar to **Bucket policy created successfully** is displayed, the selected users of department B are prevented from writing and deleting the shared data.

**Step 5 Upload data.**

Users in department A can upload data through OBS Console, OBS Browser+, APIs, and SDKs. This section takes the operations on OBS Console as an example to describe how to upload data.

1. Log in to OBS Console as a user of department A.
2. In the bucket list, click the name of the bucket that stores the shared data.
3. In the navigation pane on the left, click **Objects** and then **Upload Object**.
4. In the displayed **Upload Object** dialog box, select the upload mode, storage class, and data to be uploaded.
5. Click **Upload**.

   You can click **Task Management** in the lower part of the page to view the upload progress and result.

**Step 6 Verify the permission.**

After the permission is configured, users of department B can verify the permissions using OBS Console, OBS Browser+, APIs, and SDKs. This section takes OBS Console as an example to present how to verify that users of department B can only read the shared data.

1. Log in to OBS Console as an IAM user of department B.
2. In the bucket list, click the name of the target bucket.
3. In the left navigation pane, click **Objects**. The object list is displayed.
4. Click **Download** in the row where a public data record is located.
   - If the download fails, the download permission fails to be granted. Check whether the user group permission configuration is correct.
   - If the download is successful, the download permission is granted successfully. Go to the next step.
5. Click **Upload Object**, select a file, and click **Upload**.
   - If the upload is successful, the permission configuration for preventing write and deletion by users of other departments fails. Check whether the bucket policy is correctly configured.
   - If the upload fails, the permission configuration is successful. Go to the next step.
6. Click **Delete** in the row where a public data record is located.
   - If the deletion is successful, the permission configuration for preventing write and deletion by users of other departments fails. Check whether the bucket policy is correctly configured.
   - If the deletion fails, the permission configuration is successful.

---End

### 6.4 Data Isolation from Enterprise Partners

An enterprise expects to isolate internal data from partner data. That is, partners can view only authorized buckets.

**Prerequisites**

An account ID of the partner has been obtained.

**Procedure**

Configure a bucket policy for buckets that store partner data to allow partner users to access the buckets.

**Step 1** Log in to the HUAWEI CLOUD console as an account or an enterprise administrator (an IAM user whose user group is **admin**).
Step 2 On the homepage of the HUAWEI CLOUD console, choose Service List > Storage > Object Storage Service.

Step 3 In the OBS bucket list, click the name of the target bucket.

Step 4 In the left navigation pane, choose Permissions. On the page that is displayed, click the Bucket Policies tab.

Step 5 Click Create Bucket Policy under Custom Bucket Policies.

Step 6 Create a custom bucket policy based on the following parameter settings.
- Select Customized for Policy Mode.
- Select Allow for Effect.
- Select Include and Other account for Principal, and enter the account ID of the partner.
- For Resources, enter *, which indicates that this bucket policy applies to all the objects in the current bucket.
- For Actions, select * under General from the drop-down list, which indicates that all actions related to the object can be performed.

NOTE
You can also configure one or more specified actions by referring to Actions > Actions Related to Buckets

Step 7 Click OK to create the policy.

Step 8 Verify the permission.

After the permission is granted, partner users can use OBS Browser+ to add external buckets for permission verification.

1. Log in to OBS Browser+ as a partner user.
2. In the navigation pane on the left, choose External Bucket.
3. Click Add. The dialog box for adding an external bucket is displayed. Enter the name of the authorized bucket.
4. Click OK

If the bucket is successfully added and the action configured in Step 6 can be properly performed, the permission is granted successfully.

----End

6.5 Authorizing Business Departments with Independent Resource Permissions

A company usually consists of multiple business departments, and each department requires independent data management. In this scenario, you can allocate IAM users of different roles to each department, and configure bucket policies to authorize the IAM users with independent resource permissions.
Scenario Assumption

Assume that a company has two business departments: A and B. Each department needs a separate bucket to store data, and users of each department have the permission to upload data to their own department's bucket.

Figure 6-7 shows the logical relationships among administrators, users, and buckets between the two departments.

Figure 6-7 Logical relationship

- **Enterprise account**
- **Department A admin**
  - Department A user
  - Create a bucket and configure bucket policies.
  - Upload data.
- **Department A's bucket: dept-a**
- **Department B admin**
  - Department B user
  - Create a bucket and configure bucket policies.
  - Upload data.
- **Department B's bucket: dept-b**

**NOTE**

This example describes how to configure the upload permission for users of a department. You can configure other permissions based on the site requirements. For details about bucket policy permissions, see [Bucket Policy and Object Policy](#).

Solution and Process

The administrators of department A and department B can configure bucket policies to allow only users of their own department to upload data to their own department's bucket. For details about the configuration process, see Figure 6-8.
Figure 6-8 Permission control process

Prerequisites

You have an enterprise account of the company.

Procedure

Step 1 Create an administrator for each department and create users.

You need to use the enterprise account of the company to create IAM users as administrators and common users. A department administrator can also create common users. In this example, each department has an administrator and several users.

Add the administrator to the admin user group, which has the permissions to create users and buckets and configure bucket policies. Other users only need the permission to list buckets under the account but not permissions to create users or buckets or configure bucket policies. Therefore, add other users to user groups with the OBS Buckets Viewer permissions. For details about permissions, see Permissions Management.

1. Create a department administrator and some IAM users. For details, see Creating an IAM User.

2. Add the administrator to the admin user group, and add other users to user groups with the OBS Buckets Viewer permissions. For details, see Assigning Permissions to an IAM User.
**Step 2 Create a bucket.**

The administrator of department A creates a bucket for its own department, so does the administrator of department B.

1. Use the administrators of the two departments to log in to OBS Console in sequence.
2. On the console homepage, choose **Service List** > **Storage** > **Object Storage Service** to access OBS Console.
3. In the navigation pane on the left, choose **Object Storage**. On the page that is displayed, click **Create Bucket** in the upper right corner.
4. Select a region, storage class, and bucket policy, and then enter the bucket name.

   **NOTE**

   To ensure data security, set **Bucket Policy** to **Private** and set other parameters as prompted.

5. Click **Create Now**. The bucket is created.

**Step 3 Authorize users with the upload permission.**

The two administrators configure the upload permission for their own department users in their own bucket separately.

1. Use the administrators of the two departments to log in to OBS Console in sequence.
2. On the console homepage, choose **Service List** > **Storage** > **Object Storage Service** to access OBS Console.
3. In the navigation pane on the left, choose **Object Storage**. In the bucket list, click the department's bucket to go to the **Summary** page of the bucket.
4. In the left navigation pane, choose **Permissions**. On the page that is displayed, click the **Bucket Policies** tab.
5. Click **Create Bucket Policy** under **Custom Bucket Policies**.
6. Create a custom bucket policy by referring to **Table 6-6**.
**Figure 6-9** Settings of a bucket policy that allows specified users to list objects in the bucket

Create Bucket Policy  Learn how to configure.

<table>
<thead>
<tr>
<th>Policy Mode</th>
<th>Read-only</th>
<th>Read and write</th>
<th>Customized</th>
</tr>
</thead>
<tbody>
<tr>
<td>Provides users with customized operation permissions for the bucket and objects in the bucket.</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Effect</td>
<td>Allow</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Principal</td>
<td>Include</td>
<td>Exclude</td>
<td></td>
</tr>
<tr>
<td>Current account</td>
<td>Other account</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Username</td>
<td>poc-user1</td>
<td>poc-user2</td>
<td></td>
</tr>
<tr>
<td>Resources</td>
<td>Include</td>
<td>Exclude</td>
<td></td>
</tr>
<tr>
<td>Resource Name</td>
<td>Enter resource names.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Actions</td>
<td>Include</td>
<td>Exclude</td>
<td></td>
</tr>
<tr>
<td>Action Name</td>
<td>ListBucket</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Table 6-6** Settings of a bucket policy that allows specified users to list objects in the bucket

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Configuration</th>
</tr>
</thead>
<tbody>
<tr>
<td>Policy Mode</td>
<td>Select <strong>Customized</strong>.</td>
</tr>
<tr>
<td>Effect</td>
<td>Select <strong>Allow</strong>.</td>
</tr>
<tr>
<td>Principal</td>
<td>Select <strong>Include</strong> and <strong>Current account</strong>, and then select a user from the drop-down list. Select users of the department, to whom you want to authorize the upload permission.</td>
</tr>
<tr>
<td>Resources</td>
<td>Select <strong>Include</strong>. If the <strong>Resource Name</strong> is left blank, the policy takes effect on the entire bucket.</td>
</tr>
<tr>
<td>Actions</td>
<td>Select <strong>Include</strong> and select <strong>ListBucket</strong> from the drop-down list of the <strong>Action Name</strong>.</td>
</tr>
</tbody>
</table>

7. Click **OK**. If a message similar to **Bucket policy created successfully** is displayed, the permission is successfully configured.
8. Again, click **Create Bucket Policy** under **Custom Bucket Policies**.
9. Create another custom bucket policy by referring to **Table 6-7**.
Figure 6-10 Settings of a bucket policy that allows specified users to upload data to the bucket

Create Bucket Policy Learn how to configure.

<table>
<thead>
<tr>
<th>Policy Mode</th>
<th>Read-only</th>
<th>Read and write</th>
<th>Customized</th>
</tr>
</thead>
<tbody>
<tr>
<td>Effect</td>
<td>Allow</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Principal</td>
<td>Include</td>
<td>Exclude</td>
<td></td>
</tr>
<tr>
<td>Username</td>
<td>puc-user1</td>
<td>puc-user2</td>
<td></td>
</tr>
<tr>
<td>Resources</td>
<td>Include</td>
<td>Exclude</td>
<td></td>
</tr>
<tr>
<td>Resource Name</td>
<td>*</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Actions</td>
<td>Include</td>
<td>Exclude</td>
<td></td>
</tr>
<tr>
<td>Action Name</td>
<td>PutObject</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Table 6-7 Settings of a bucket policy that allows specified users to upload data to the bucket

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Configuration</th>
</tr>
</thead>
<tbody>
<tr>
<td>Policy Mode</td>
<td>Select Customized.</td>
</tr>
<tr>
<td>Effect</td>
<td>Select Allow.</td>
</tr>
<tr>
<td>Principal</td>
<td>Select Include and Current account, and then select a user from the drop-down list. Select users of the department, to whom you want to authorize the upload permission.</td>
</tr>
</tbody>
</table>
| Resources   | Select Include and enter the resource name based on the scope of the data to be uploaded.  
- * indicates that data can be uploaded to any place in the bucket.  
- If data can only be uploaded to only one or several folders in the bucket, enter the folder path and put a * to its end (for example, example-folder/*). Use commas (,) to separate one resource name from another. |
| Actions     | Select Include and select PutObject from the drop-down list of Action Name. |
10. Click **OK**. If a message similar to **Bucket policy created successfully** is displayed, the permission is successfully configured.

**Step 4 Verify the permission.**

After the permission is configured, users of department A and department B can verify the permissions by uploading objects through OBS Console, OBS Browser+, APIs, and SDKs.

The permission verification should focus on the following aspects (taking department A for an example):

1. Users of department A can successfully upload objects to the bucket of department A.
   - If users are allowed to upload objects to only the specified folder, ensure that:
     a. Objects can be successfully uploaded to the specified folder.
     b. Upload of objects to folders other than the specified one will fail.
2. Users of department A fail to upload objects to the bucket of department B.
3. Users of department A fail to download or delete any object from the bucket of department A.
4. Users of department A fail to download or delete any object from the bucket of department B.

If the preceding requirements are met, the permission configuration is successful.

---End

### 6.6 Isolating Bucket Resources Between Business Departments

According to the permission control configured in **Authorizing Business Departments with Independent Resource Permissions**, users in different departments can only access resources of their own departments. However, they can read all bucket resources under the enterprise account. This section describes how to use OBS Browser+ to isolate bucket resources between business departments by adding external buckets.

**Scenario Assumption**

Assume that a company has two business departments: A and B. Each department needs a separate bucket to store data, and users of each department can view and upload data to only their own department’s bucket.

**Figure 6-11** shows the logical relationships among administrators, users, and buckets between the two departments.
This example describes how to configure the upload permission for users of a department. You can configure other permissions based on the site requirements. For details about bucket policy permissions, see Bucket Policy and Object Policy.

Solution and Process

This solution should focus on the following aspects:

1. Do not grant OBS access permissions to users created by a department administrator.
2. Configure a bucket policy that allows users of their own department to perform list operations only in their own bucket.
3. Configure a bucket policy that allows users of their own department to upload objects only to their own bucket.

Figure 6-12 shows the process.
**Prerequisites**

You have an enterprise account of the company.

**Procedure**

**Step 1** Create an administrator for each department and create users.

You need to use the enterprise account of the company to create IAM users as administrators and common users. A department administrator can also create common users. In this example, each department has an administrator and several users.

Add the administrator to the **admin** user group, which has the permissions to create users and buckets and configure bucket policies. In this example, you do not need to log in to the IAM console and grant common users of the department with any OBS permissions. For details about permissions, see Permissions Management.
1. Create a department administrator and some IAM users. For details, see Creating an IAM User.
2. Add the administrator to the admin user group. Do not add other users to user groups with OBS access permissions. For details, see Assigning Permissions to an IAM User.

Step 2 Create a bucket.

The administrator of department A creates a bucket for its own department, so does the administrator of department B.

1. Use the administrators of the two departments to log in to OBS Console in sequence.
2. On the console homepage, choose Service List > Storage > Object Storage Service to access OBS Console.
3. In the navigation pane on the left, choose Object Storage. On the page that is displayed, click Create Bucket in the upper right corner.
4. Select a region, storage class, and bucket policy, and then enter the bucket name.

   **NOTE**

   To ensure data security, set Bucket Policy to Private and set other parameters as prompted.

5. Click Create Now. The bucket is created.

Step 3 Authorize users with the list permission.

The two administrators configure the list permission for their own department users in their own bucket separately.

1. Use the administrators of the two departments to log in to OBS Console in sequence.
2. On the console homepage, choose Service List > Storage > Object Storage Service to access OBS Console.
3. In the navigation pane on the left, choose Object Storage. In the bucket list, click the department’s bucket to go to the Summary page of the bucket.
4. In the left navigation pane, choose Permissions. On the page that is displayed, click the Bucket Policies tab.
5. Click Create Bucket Policy under Custom Bucket Policies.
6. Create a custom bucket policy by referring to Table 6-8.
Figure 6-13 Settings of a bucket policy that allows specified users to perform all list operations

Table 6-8 Settings of a bucket policy that allows specified users to perform all list operations

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Configuration</th>
</tr>
</thead>
<tbody>
<tr>
<td>Policy Mode</td>
<td>Select Customized.</td>
</tr>
<tr>
<td>Effect</td>
<td>Select Allow.</td>
</tr>
<tr>
<td>Principal</td>
<td>Select Include and Current account, and then select a user from the drop-down list. Select users of the department, to whom you want to authorize the read permission.</td>
</tr>
<tr>
<td>Resources</td>
<td>Select Include. If the Resource Name is left blank, the policy takes effect on the entire bucket.</td>
</tr>
<tr>
<td>Actions</td>
<td>Select Include and select List* from the drop-down list of Action Name.</td>
</tr>
</tbody>
</table>

7. Click OK. If a message similar to Bucket policy created successfully is displayed, the permission is successfully configured.

Step 4 Authorize users with the upload permission.

The two administrators configure the upload permission for their own department users in their own bucket separately.

1. Use the administrators of the two departments to log in to OBS Console in sequence.
2. On the console homepage, choose Service List > Storage > Object Storage Service to access OBS Console.

3. In the navigation pane on the left, choose Object Storage. In the bucket list, click the department’s bucket to go to the Summary page of the bucket.

4. In the left navigation pane, choose Permissions. On the page that is displayed, click the Bucket Policies tab.

5. Click Create Bucket Policy under Custom Bucket Policies.

6. Create a custom bucket policy by referring to Table 6-9.

**Figure 6-14** Settings of a bucket policy that allows specified users to upload data to the bucket

![Create Bucket Policy](image)

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Configuration</th>
</tr>
</thead>
<tbody>
<tr>
<td>Policy Mode</td>
<td>Select Customized.</td>
</tr>
<tr>
<td>Effect</td>
<td>Select Allow.</td>
</tr>
<tr>
<td>Principal</td>
<td>Select Include and Current account, and then select a user from the drop-down list. Select users of the department, to whom you want to authorize the upload permission.</td>
</tr>
</tbody>
</table>

**Table 6-9** Settings of a bucket policy that allows specified users to upload data to the bucket
**Parameter** | **Configuration**
---|---
Resources | Select **Include** and enter the resource name based on the scope of the data to be uploaded.  
- * indicates that data can be uploaded to any place in the bucket.  
- If data can only be uploaded to only one or several folders in the bucket, enter the folder path and put a * to its end (for example, *example-folder/*). Use commas (,) to separate one resource name from another.
Actions | Select **Include** and select **PutObject** from the drop-down list of **Action Name**.

7. Click **OK**. If a message similar to **Bucket policy created successfully** is displayed, the permission is successfully configured.

**Step 5 Verify the permission.**

After the permission is configured, users of department A and department B can verify the permission through OBS Browser+.

---

**NOTE**

Users in the two departments have only the permission to access a specified bucket. Therefore, it is normal that these users are prompted that their access is restricted when logging in to OBS Console.

In this case, use OBS Browser+ to add the bucket of your own department to OBS Browser+ as an external bucket for permission verification and subsequent upload operations.

**To verify the permission on OBS Browser+, perform the following steps:**

1. **Download OBS Browser+**.
2. **Obtain the access keys (AK and SK) of the department users**.
3. **Add an account and use the account to log in to OBS Browser**.

**Table 6-10 Parameters for adding an account**

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
</table>
| Account Name| Account names on OBS Browser+ are used to distinguish one login user from another, which are irrelevant to your registered cloud service account or IAM usernames.  
For example, when adding a user of department A, the account name does not need to be the same as the IAM username.  
OBS Browser+ identifies the registered account or IAM user based on the AK and SK. |
<p>| Service     | HUAWEI CLOUD OBS (default)                                                  |</p>
<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Access Key ID</td>
<td>AK obtained in step 5</td>
</tr>
<tr>
<td>Secret Access Key</td>
<td>SK obtained in step 5</td>
</tr>
<tr>
<td>Access Path</td>
<td>Leave this parameter blank.</td>
</tr>
<tr>
<td>Remember my secret access key</td>
<td>This check box is selected by default.</td>
</tr>
</tbody>
</table>

**NOTE**

Due to the preceding permission configuration, it is normal that the system displays a message indicating that the access is restricted after a department user logs in to OBS Browser+.

4. In the navigation pane on the left, choose **External Bucket**.
5. Click **Add**. The dialog box for adding an external bucket is displayed. Enter the name of the authorized bucket.

**Figure 6-15 Adding an external bucket**

6. Click **OK**. The external bucket is displayed in the bucket list.
7. Upload a file to the bucket and verify the upload permission.

**The permission verification should focus on the following aspects (taking department A for an example):**

1. When users in department A log in to OBS Browser+ for the first time, a message is displayed indicating that the access is restricted and no bucket is displayed.
2. Users of department A can successfully add the bucket of department A on OBS Browser+.
3. Users of department A fail to add the bucket of department B.
4. Users of department A can successfully upload objects to the bucket of department A.
   - If users are allowed to upload objects to only the specified folder, ensure that:
     a. Objects can be successfully uploaded to the specified folder.
     b. Upload of objects to folders other than the specified one will fail.
5. Users of department A fail to download or delete any object from the bucket of department A.

If the preceding requirements are met, the permission configuration is successful.

----End
7 Verifying Data Consistency

7.1 Overview

Background

Data inconsistency may occur due to network hijacking, cache, and other reasons during object upload and download.

Solution

OBS verifies data consistency by calculating the MD5 value when data is uploaded or downloaded. By default, OBS does not automatically verify data consistency. You can enable data consistency verification when uploading and downloading objects. Methods are explained in the following table.

### NOTE

- The consistency verification methods are compatible with each other. Specifically, you can use one method to verify data consistency when uploading objects, and use another method to verify data consistency during download of the objects.
- During object download, the MD5 verification takes effect only when the object to be downloaded has an MD5 value.

<table>
<thead>
<tr>
<th>Method</th>
<th>Description</th>
<th>Operation Guide</th>
</tr>
</thead>
<tbody>
<tr>
<td>obsutil</td>
<td>With obsutil, a command line tool, you can run a simple command to upload or download objects and specify whether to enable MD5 verification.</td>
<td>Using obsutil to Verify Data Consistency During Upload</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Using obsutil to Verify Data Consistency During Download</td>
</tr>
<tr>
<td>Method</td>
<td>Description</td>
<td>Operation Guide</td>
</tr>
<tr>
<td>------------</td>
<td>-------------------------------------------------------------------------------------------------</td>
<td>----------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>OBS Browser</td>
<td>With OBS Browser, a GUI-based tool, you can enable or disable MD5 verification through a few clicks. OBS Browser also provides task management that allows you to view the verification status.</td>
<td>Using OBS Browser to Verify Data Consistency During Upload</td>
</tr>
<tr>
<td>OBS Browser</td>
<td></td>
<td>Using OBS Browser to Verify Data Consistency During Download</td>
</tr>
<tr>
<td>OBS SDK</td>
<td>If you are a developer, you can use the OBS SDK to conduct secondary development for MD5 verification, and then proceed with the verification result according to your service needs.</td>
<td>Using OBS SDKs to Verify Data Consistency During Upload</td>
</tr>
<tr>
<td>OBS SDK</td>
<td></td>
<td>Using OBS SDKs to Verify Data Consistency During Download</td>
</tr>
</tbody>
</table>

When an object is being uploaded, OBS calculates the MD5 value of the object on the client before uploading the object. Then OBS server calculates the MD5 value of the uploaded object and compares it with the MD5 value carried in the upload request. If the two values are consistent, the upload is successful. If the two values are inconsistent, the upload fails. Figure 7-1 illustrates how to use the MD5 value to verify data consistency during upload.
When an object is being downloaded, OBS calculates the MD5 value of the downloaded object and compares it with its original MD5 value. If the two values are consistent, the download is successful. If the two values are inconsistent, the download fails. Figure 7-2 illustrates how to use the MD5 value to verify data consistency during download.
7.2 Verifying Data Consistency During Upload

obsutil, OBS Browser, and OBS SDKs support verification of data consistency during upload. You can select a method that meets your requirements. This topic provides guidance on how to use these methods to verify data consistency during object upload.

**Using obsutil to Verify Data Consistency During Upload**

obsutil supports data consistency verification through the additional parameter `vmd5`.

For example, to upload the `test.txt` file drive D on a Windows OS to bucket `mytestbucket`, run the following command to enable consistency verification:

```bash
obsutil cp D:\test.txt obs://mytestbucket/test.txt -vmd5
```

The object is uploaded after the verification passes, and **Upload successfully** is displayed in the command output.
Using OBS Browser to Verify Data Consistency During Upload

By default, MD5 verification is disabled on OBS Browser. To enable MD5 verification and upload objects through OBS Browser, perform the following steps:

**Step 1** Log in to OBS Browser.

**Step 2** Click in the upper right corner of the page and choose System Configuration.

**Step 3** Choose System Configuration > Basic Configuration. In the dialog box that is displayed, select Enable MD5 verification, as shown in Figure 7-3.
**Figure 7-3 Configuring MD5 verification**

![System Configuration](image)

**Step 4** Click **Save**.

**Step 5** Select the bucket to which the file is to be uploaded and upload the file.

In the upper right corner of the page, click ![Cloud](image). Then in the running task list, you can view the file uploading status and MD5 verification status. For details, see **Figure 7-4**.

**Figure 7-4 MD5 verification statuses**

![Running and Completed Tasks](image)

**Step 6** After the file is uploaded, you can view the MD5 verification result in the completed task list, as shown in **Figure 7-5**.
Using OBS SDKs to Verify Data Consistency During Upload

OBS provides SDKs for multiple programming languages such as Java and Python. You can specify the `Content-MD5` parameter to enable consistency verification when uploading objects through any OBS SDK. For details about how to compute and set the MD5 value of an object, see section "Setting Object Properties" in OBS SDK of the programming language that you use.

For example, to upload file `text.txt` from drive D on a Windows OS to bucket `mytestbucket` by using OBS Java SDK, you can use the following sample code to verify data consistency through the MD5 value:

```java
String endPoint = "https://your-endpoint";
String ak = "*** Provide your Access Key ***";
String sk = "*** Provide your Secret Key ***";
// Create an instance of ObsClient.
ObsClient obsClient = new ObsClient(ak, sk, endPoint);
// Compute and set the MD5 value.
ObjectMetadata metadata = new ObjectMetadata();
File file = new File("D:\text.txt");
FileInputStream fis = new FileInputStream(file);
InputStream is = (InputStream)fis;
String contentMd5 = obsClient.base64Md5(is);
metadata.addUserMetadata("contentMd5", contentMd5);
// Upload a file that has the MD5 value.
obsClient.putObject("mytestbucket", "text.txt", file, metadata);
```

- The MD5 value of an object must be a Base64-encoded digest.
- The OBS server will compare this MD5 value with the MD5 value obtained by object data calculation. If the two values are not the same, the upload fails and the HTTP status code 400 is returned. If the two values are consistent, the object is successfully uploaded and the HTTP status code 200 is returned.

7.3 Verifying Data Consistency During Download

OBS Browser, obsutil, and OBS SDKs support consistency verification during object download. You can select a verification method that meets your requirements. This topic provides guidance on how to use these methods to verify data consistency during object download.

Prerequisites

The object to be downloaded has an MD5 value. Data consistency is not verified, if the object does not have an MD5 value. The MD5 value of an object needs to be computed and set when the object is uploaded. For details, see [Verifying Data Consistency During Upload](#).
Using obsutil to Verify Data Consistency During Download

obsutil supports data consistency verification through the additional parameter `vmd5`.

For example, to download the `test.txt` file from bucket `mytestbucket` to a Windows PC, perform the following steps to enable data consistency verification:

**Step 1** Run the following command to check whether the object to be downloaded has MD5 information:

```bash
obsutil stat obs://test-bucket/test.txt
```

- If the returned information contains MD5 information, as shown in the following figure, go to **Step 2**.
  ```bash
d:\obsutil\windows_end64\obsutil stat obs://mytestbucket/test.txt
Start at 2019-06-10 09:08:00.9978182 +0000 UTC

Key:  
  obs://mytestbucket/test.txt
LastModified:  
  2019-06-10T09:04:25Z
Size:  
  291
StorageClass:  
  standard
MDS:  
  2b0ec156a66e0035ea18d85c1b3ce619
Etag:  
  2b0ec156a66e0035ea18d85c1b3ce619
ContentType:  
  text/plain
```

- If MD5 information is not contained, consistency verification cannot be performed when the object is downloaded.

**Step 2** Run the following command to download the object:

```bash
obsutil cp obs://mytestbucket/test.txt D:\test.txt -vmd5
```

- The message highlighted in the following figure is displayed in the command output when the object is downloaded and the consistency verification passes.
  ```bash
  Download successfully.
  317B, a6d2a254f93af83e6e6e5232bb4e0, obs://mytestbucket/text.txt --> D:\text.txt, cost [50], status [200], request id [000001664466E8C38D8F29740856e9]
  ```

- If the object does not have an MD5 value, the object can be successfully downloaded but data consistency is not verified. The message highlighted in the following figure is displayed in the command output:
  ```bash
  Download successfully.
  317B, n/a, obs://mytestbucket/text.txt --> D:\text.txt, cost [100], status [200], request id [000001664466E8C38D8F29740856e9]
  Warn: Cannot get the valid md5 value of key [text.txt] in bucket [mytestbucket] to check
  ```

----End
Using OBS Browser to Verify Data Consistency During Download

By default, MD5 verification is disabled on OBS Browser. To enable MD5 verification and download objects from OBS Browser, perform the following steps:

**Step 1** Log in to OBS Browser.

**Step 2** Click in the upper right corner of the page and choose System Configuration.

**Step 3** In the System Configuration > Basic Configuration dialog box, select Enable MD5 verification. For details, see Figure 7-6.

![Figure 7-6 Configuring MD5 verification](image)

**Step 4** Click Save.

**Step 5** Select the bucket from which you want to download the object and download the object.

In the upper right corner of the page, click . Then in the running task list, you can view the object downloading status and MD5 verification status. For details, see Figure 7-7.
Step 6  After the object is downloaded, you can view the MD5 verification result in the list of completed tasks. For details, see Figure 7-8.

Using OBS SDKs to Verify Data Consistency During Download

The verification compares the MD5 value in the custom metadata of the object to be downloaded with that of the downloaded object to check whether the downloaded object is consistent with the original object.

For example, to download file text.txt from bucket mytestbucket to drive D on a Windows OS by using OBS Java SDK, you can use the following sample code to verify data consistency through the MD5 value:

```java
String endPoint = "https://your-endpoint";
String ak = "*** Provide your Access Key ***";
String sk = "*** Provide your Secret Key ***";
// Create an instance of ObsClient.
final ObsClient obsClient = new ObsClient(ak, sk, endPoint);
// Obtain the MD5 value of the object.
ObjectMetadata metadata = obsClient.getObjectMetadata("mytestbucket", "test.txt");
String md5Origin = metadata.getUserMetadata("contentMd5");
// Compute the MD5 value of the downloaded object.
Obsobject obsobject = obsClient.getObject("mytestbucket", "test.txt");
String md5Download = obsClient.base64Md5(obsobject.getObjectContent());
// Compare the MD5 values.
if(md5Origin.contentEquals(md5Download))
    System.out.println("Object MD5 validation passes!\n");
else
    System.out.println("Object MD5 validation failed!\n");
```

In the preceding sample code, the contentMd5 parameter used to obtain the MD5 value of an object is a customized metadata specified when the object is uploaded. Replace it with the one that you customized for the actual development.
Performance Optimization

OBS manages partitions based on the UTF-8 code range of object names and implements horizontal expansion and dynamic load balancing accordingly. If you use sequential prefixes (such as timestamps or alphabetical order) in object naming, object access requests may be concentrated in a specific partition, causing access hotspots. The request rate in a hotspot partition is limited, as a result, access delay increases.

**Random prefixes for object naming are recommended so that requests are evenly distributed across partitions, allowing horizontal expansion.**

**Example:**

In a typical scenario of log archiving, the names of objects to be uploaded are as follows:

<table>
<thead>
<tr>
<th>YourBucket/Obslog/20190610-01.log.tar.gz</th>
</tr>
</thead>
<tbody>
<tr>
<td>YourBucket/Obslog/20190610-02.log.tar.gz</td>
</tr>
<tr>
<td>YourBucket/Obslog/20190610-03.log.tar.gz</td>
</tr>
<tr>
<td>YourBucket/Obslog/20190610-04.log.tar.gz</td>
</tr>
</tbody>
</table>

...  
| YourBucket/Obslog/20190611-01.log.tar.gz |
| YourBucket/Obslog/20190611-02.log.tar.gz |
| YourBucket/Obslog/20190611-03.log.tar.gz |
| YourBucket/Obslog/20190611-04.log.tar.gz |

You are advised to add a hexadecimal hash prefix with three or more digits to the object name.

<table>
<thead>
<tr>
<th>YourBucket/6ac-Obslog/20140610-01.log.tar.gz</th>
</tr>
</thead>
<tbody>
<tr>
<td>YourBucket/b42-Obslog/20140610-02.log.tar.gz</td>
</tr>
<tr>
<td>YourBucket/17f-Obslog/20140610-03.log.tar.gz</td>
</tr>
<tr>
<td>YourBucket/ac9-Obslog/20140610-04.log.tar.gz</td>
</tr>
</tbody>
</table>

...  
| YourBucket/95d-Obslog/20140611-01.log.tar.gz |
| YourBucket/4a5-Obslog/20140611-02.log.tar.gz |
| YourBucket/ea2-Obslog/20140611-03.log.tar.gz |
| YourBucket/ba3-Obslog/20140611-04.log.tar.gz |
9 Using a Presigned URL to Access OBS

Context

OBS is widely used as the storage for web applications and mobile apps (Android and iOS). When accessing OBS from Android or iOS apps, do not use fixed access keys (AK and SK), which may be cracked by hacker software, and as a result, data stored in the cloud storage may be stolen or even tampered with. To better protect application data and prevent data leakage and unauthorized access after attacks, you are advised to use presigned URLs to access OBS.

Principles

Each request initiated by an application client applies for a presigned URL from the application server. The validity period of the presigned URL is determined by the application server. Figure 9-1 details the process.

Figure 9-1 Process for a mobile app to access data in OBS
Analysis on roles:

- Application client: A mobile app of an end user. An application client applies for presigned URLs from the application server, and accesses OBS to upload or download data.
- Application server: Background services provided by developers of such Android/iOS apps. Application servers manage credential information and issue presigned URLs.
- OBS: HUAWEI CLOUD Object Storage Service, which processes data requests from mobile apps.

Implementation procedure:

1. An application client applies for a presigned URL from the application server. Access keys (AK and SK) are not required for accessing OBS from Android or iOS apps. But a presigned URL must be obtained from the application server before accessing data in OBS, and required information must be carried in the URL, including the request type, resource path, and resource name. For example, an upload request needs to indicate that the URL is for uploading data. In the URL, the upload path and object name are specified. Similarly, a URL for downloading data should contain the name of the object to be downloaded.

2. As a trusted device, the application server stores access keys (AK and SK). After verifying that the client is valid, the application server generates a presigned URL using the permanent access keys (AK and SK), in accordance with the operation type and resources to be accessed by the client. A sample URL:

   ```plaintext
   https://examplebucket.obs.cn-north-4.myhuaweicloud.com/objectkey?AccessKeyId=AccessKeyID&Expires=1532779451&Signature=0Akylf43Bm3mD1bh2rM3dmVp1Bo%3D
   ```

3. Android/iOS mobile apps obtain the URL and use the URL to perform desired operations, such as uploading and downloading. The URL contains the access key ID (AK) of the user, signature, validity period, and resource information. Anyone who has the URL can perform the operation. After receiving the request and verifying the signature, OBS deems that the request is executed by the user who issues the URL. For example, if the URL of an object download request carrying signature information is constructed, the user who obtains the URL can download the object, but the URL is valid only within the expiration time specified by the parameter of `Expires`. The URL that carries the signature is used to allow others to use the presigned URL for identity authentication when the SK is not provided, and perform the predefined operation.

Prerequisites

1. A bucket has been created.
   Create a bucket on OBS Console. Configure the bucket permissions, and allow it to be read/written privately, read publicly, or written privately. For details, see Creating a Bucket and Configuring a Custom Bucket Policy.

2. Permanent access keys (AK and SK) have been obtained.
   The presigned URL is generated using the permanent access keys. For details about how to obtain access keys, see Obtaining Access Keys (AK/SK). The user who uses the access keys (AK/SK) needs to have the minimum required...
permissions. For details about how to authorize the permissions, see Creating an IAM User and Granting the User the Access Permissions to OBS Resources.

Algorithms for Computing Presigned URLs

For details, see Authentication Signature in a URL.

Procedure

Step 1 Configure an application server.

1. Obtain the SDK.
   Obtain the required SDK from the Java SDK Reference.
2. Generate the code for issuing a presigned URL.
   The following example describes how to use the Java language for development on the application server.

   ```java
   //Endpoint of the requested bucket
   String endPoint = "http://your-endpoint";

   //Provide your AK and SK.
   String ak = "**** Provide your Access Key ****";
   String sk = "**** Provide your Secret Key ****";

   //Create an ObsClient
   ObsClient obsClient = new ObsClient(ak, sk, endPoint);

   //Define the expiration time, in seconds.
   long expireSeconds = 3600L;

   //Specify the requested operation.
   TemporarySignatureRequest request = new TemporarySignatureRequest(HttpMethodEnum.PUT, expireSeconds);

   //Specify the bucket name and object name involved in this operation.
   request.setBucketName("bucketname");
   request.setObjectKey("objectname");

   TemporarySignatureResponse response = obsClient.createTemporarySignature(request);

   //If the following message is returned, the presigned URL is successfully issued, and you can print the URL information.
   System.out.println(response.getSignedUrl());
   ```

   For more information and code samples, see Using a URL for Authorized Access.

Step 2 Use the presigned URL to initiate an OBS access request.

```java
public class Demo extends Activity{

    private static String bucketName = "my-obs-bucket-demo";
    private static String objectKey = "my-obs-object-key-demo";
    private static OkHttpClient httpClient;
```
private static StringBuffer sb;

@Override
protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.activity_main);
    sb = new StringBuffer();

    /* Constructs a client instance with your account for accessing OBS */
    httpClient = new OkHttpClient.Builder().followRedirects(false).retryOnConnectionFailure(false)
        .cache(null).build();

    final TextView tv = (TextView)findViewById(R.id.tv);
    tv.setText("Click to start test");
    tv.setOnClickListener(new View.OnClickListener() {
        @Override
        public void onClick(View v) {
            tv.setClickable(false);
            AsyncTask<Void, Void, String> task = new DownloadTask();
            task.execute();
        }
    });
}

class DownloadTask extends AsyncTask<Void, Void, String> {
    @Override
    protected String doInBackground(Void... params) {
        try {
            /* You need to construct an object upload request and send it to the application server to generate a presigned URL for accessing OBS. */
            /* If the response result is stored in response, obtain the URL using the getSignedUrl() method. */
            sb.append("Uploading a new object to OBS from a file

            Request.Builder builder = new Request.Builder();
            //Make a PUT request to upload an object.
            Request httpRequest = builder.url(response.getSignedUrl()).put(RequestBody.create(MediaType.parse(contentType), "Hello OBS").build();
            Call c = httpClient.newCall(httpRequest);
            Response res = c.execute();
            sb.append("\tStatus:" + res.code());
            if (res.body() != null) {
                sb.append("\tContent:" + res.body().string() + "\n");
            }
            res.close();
            /* You need to construct an object download request and send it to the application server to generate a presigned URL for accessing OBS. */
            /* If the response result is stored in response, obtain the URL using the getSignedUrl() method. */
            sb.append("Downloading an object

            Request.Builder builder = new Request.Builder();
            //Make a GET request to download an object.
            Request httpRequest = builder.url(response.getSignedUrl()).get().build();
            Call c = httpClient.newCall(httpRequest);
            Response res = c.execute();
            System.out.println("\tStatus:" + res.code());
            if (res.body() != null) {
                sb.append("\tContent:" + res.body().string() + "\n");
            }
        }
    }

    @Override
    public void onPostExecute(String result) {
        super.onPostExecute(result);
        tv.setText(sb.toString());
    }
}

public class Main {
    public static void main(String[] args) {
        Activity activity = new Activity();
        activity.setContentView(R.layout.activity_main);
        activity.findViewById(R.id.tv).setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                tv.setClickable(false);
                new DownloadTask().execute();
            }
        });
    }
}
```java
res.close();
return sb.toString();
}
catch (Exception e)
{
    sb.append("\n\n");
    sb.append(e.getMessage());
    return sb.toString();
}
finally
{
    if (httpClient != null)
    {
        try
        {
            /*
             * Close obs client
             */
            httpClient.close();
        }
        catch (IOException e)
        {
        }
    }
}

@Override
protected void onPostExecute(String result)
{
    TextView tv = (TextView) findViewById(R.id.tv);
    tv.setText(result);
    tv.setOnClickListener(null);
    tv.setMovementMethod(ScrollingMovementMethod.getInstance());
}
}

----End
```
<table>
<thead>
<tr>
<th>Release Date</th>
<th>What's New</th>
</tr>
</thead>
</table>
| 2020-03-24   | This issue is the seventh official release. This issue incorporates the following changes:  
● Added section "Verifying Data Consistency". |
| 2019-12-14   | This issue is the sixth official release. This issue incorporates the following change:  
● Added the section "Using a Presigned URL to Access OBS". |
| 2019-11-05   | This issue is the fifth official release. This issue incorporates the following change:  
● Added the topic "Isolating Bucket Resources Between Business Departments" to the "Enterprise Data Access Control" section. |
| 2019-10-09   | This issue is the fourth official release. This issue incorporates the following change:  
● Added the topic "Authorizing Business Departments with Independent Resource Permissions" to the "Enterprise Data Access Control" section. |
| 2019-07-06   | This issue is the third official release. This issue incorporates the following change:  
● Added the section "Performance Optimization". |
| 2018-11-30   | This issue is the second official release. This issue incorporates the following change:  
● Added the section "Overview of OBS Best Practices." |
| 2018-09-30   | This issue is the first official release. |