



## Object Storage Service

# User Guide

Date 2024-01-29

---

# Contents

---

<b>1 Service Overview.....</b>	<b>1</b>
1.1 About OBS.....	1
1.2 Advantages.....	4
1.3 Application Scenarios.....	5
1.4 Permissions Management.....	9
1.5 Restrictions and Limitations.....	15
1.6 Related Services.....	18
1.7 Basic Concepts.....	19
1.7.1 Objects.....	19
1.7.2 Buckets.....	20
1.7.3 Parallel File System.....	21
1.7.4 Access Keys (AK/SK).....	21
1.7.5 Endpoints and Domain Names.....	22
1.7.6 Region and AZ.....	24
<b>2 OBS Console Operation Guide.....</b>	<b>25</b>
2.1 Console Function Overview.....	25
2.2 Restrictions.....	26
2.3 Getting Started.....	27
2.3.1 Process Description.....	27
2.3.2 Configuring User Permissions.....	28
2.3.3 Creating a Bucket.....	30
2.3.4 Uploading an Object.....	33
2.3.5 Downloading an Object.....	35
2.3.6 Deleting an Object.....	35
2.3.7 Deleting a Bucket.....	36
2.4 Storage Classes Overview.....	37
2.5 Managing Buckets.....	38
2.5.1 Creating a Bucket.....	38
2.5.2 Viewing Basic Information of a Bucket.....	42
2.5.3 Searching for a Bucket.....	45
2.5.4 Deleting a Bucket.....	46
2.6 Managing Objects.....	47
2.6.1 Creating a Folder.....	47

2.6.2 Uploading an Object.....	48
2.6.3 Downloading an Object.....	51
2.6.4 Sharing an Object.....	52
2.6.5 Searching for an Object or Folder.....	54
2.6.6 Accessing an Object Using Its URL.....	57
2.6.7 Restoring Objects from the Cold Storage.....	58
2.6.8 Deleting an Object or Folder.....	60
2.6.9 Undeleting an Object.....	62
2.6.10 Managing Fragments.....	65
2.7 Server-Side Encryption.....	65
2.7.1 Server-Side Encryption Overview.....	65
2.7.2 Bucket Default Encryption.....	66
2.7.3 Uploading an Object in Server-Side Encryption Mode.....	68
2.8 Object Metadata.....	69
2.8.1 Object Metadata Overview.....	69
2.8.2 About Object Metadata Content-Type.....	71
2.8.3 Configuring Object Metadata.....	80
2.9 Permissions Control.....	80
2.9.1 Overview.....	80
2.9.2 Permission Control Mechanisms.....	81
2.9.2.1 IAM Policies.....	81
2.9.2.2 Bucket Policies and Object Policies.....	85
2.9.2.3 Bucket ACLs and Object ACLs.....	93
2.9.2.4 Relationship Between a Bucket ACL and a Bucket Policy.....	97
2.9.2.5 How Does Authorization Work When Multiple Access Control Mechanisms Co-Exist?.....	98
2.9.3 Bucket Policy Parameters.....	99
2.9.3.1 Effect.....	99
2.9.3.2 Principals.....	100
2.9.3.3 Resources.....	100
2.9.3.4 Actions.....	101
2.9.3.5 Conditions.....	105
2.9.4 Configuring IAM Policies.....	110
2.9.4.1 Creating an IAM User and Granting OBS Permissions.....	110
2.9.4.2 Configuring Fine-Grained Policies.....	111
2.9.4.3 OBS Resources.....	114
2.9.5 Configuring a Bucket Policy.....	114
2.9.5.1 Creating a Bucket Policy with a Template.....	115
2.9.5.2 Creating a Custom Bucket Policy (Visual Editor).....	127
2.9.5.3 Creating a Custom Bucket Policy (JSON View).....	131
2.9.6 Configuring an Object Policy.....	132
2.9.7 Configuring a Bucket ACL.....	133
2.9.8 Configuring an Object ACL.....	134

2.9.9 Application Cases.....	135
2.9.9.1 Granting an IAM User Permissions to Operate a Specific Bucket.....	136
2.9.9.2 Granting Other Accounts Permissions to Operate a Specific Bucket.....	137
2.9.9.3 Restricting Access to a Bucket for Specific Addresses.....	139
2.9.9.4 Limiting the Time When Objects in a Bucket Are Accessible.....	140
2.9.9.5 Granting Anonymous Users Permission to Access Objects.....	141
2.9.9.6 Granting Anonymous Users Permission to Access Folders.....	142
2.10 Versioning.....	143
2.10.1 Versioning Overview.....	144
2.10.2 Configuring Versioning.....	147
2.11 Logging.....	148
2.11.1 Logging Overview.....	148
2.11.2 Configuring Access Logging for a Bucket.....	151
2.12 Tags.....	153
2.12.1 Tag Overview.....	153
2.12.2 Configuring Tags for a Bucket.....	153
2.13 Event Notifications.....	154
2.13.1 SMN-Enabled Event Notifications.....	154
2.13.2 Configuring SMN-Enabled Event Notification.....	155
2.13.3 Application Example: Configuring SMN-Enabled Event Notification.....	158
2.14 Lifecycle Management.....	160
2.14.1 Lifecycle Management Overview.....	160
2.14.2 Configuring a Lifecycle Rule.....	162
2.15 Configuring User-Defined Domain Names.....	164
2.15.1 Overview.....	165
2.15.2 Configuring a User-Defined Domain Name.....	165
2.16 Static Website Hosting.....	167
2.16.1 Static Website Hosting Overview.....	167
2.16.2 Redirection Overview.....	167
2.16.3 Configuring Static Website Hosting.....	168
2.16.4 Configuring Redirection.....	174
2.16.5 Using a User-Defined Domain Name to Configure Static Website Hosting.....	176
2.17 Cross-Origin Resource Sharing.....	183
2.17.1 CORS Overview.....	183
2.17.2 Configuring CORS.....	184
2.18 URL Validation.....	187
2.18.1 URL Validation Overview.....	187
2.18.2 Configuring URL Validation.....	187
2.19 Monitoring.....	189
2.19.1 Monitoring OBS.....	189
2.19.2 OBS Monitoring Metrics.....	190
2.20 Cloud Trace Service.....	192

2.21 Task Center.....	196
2.22 Related Operations.....	196
2.22.1 Creating an IAM Agency.....	196
2.23 Troubleshooting.....	197
2.23.1 An Object Fails to Be Downloaded Using Internet Explorer 11.....	198
2.23.2 OBS Console Cannot Be Opened in Internet Explorer 9.....	198
2.23.3 The Object Name Changes After an Object with a Long Name Is Downloaded to a Local Computer .....	199
2.23.4 Failed to Configure Event Notifications.....	200
2.23.5 Time Difference Is Longer Than 15 Minutes Between the Client and Server.....	200
2.24 Error Code List.....	200
<b>3 FAQ.....</b>	<b>203</b>
3.1 OBS Basics.....	203
3.1.1 How Can I Get Started with OBS?.....	203
3.1.2 How Do I Obtain an OBS Endpoint?.....	203
3.1.3 What Are the Advantages of Object Storage over SAN and NAS Storage?.....	203
3.1.4 Which Types of Data Can Be Stored in OBS?.....	204
3.1.5 How Much Data Can I Store in OBS?.....	204
3.1.6 Does OBS Support Traffic Monitoring?.....	204
3.1.7 Can Folders in OBS Be Used the Same Way as in a File System?.....	205
3.1.8 Where Is Data Stored in OBS?.....	206
3.1.9 Does OBS Support Access over HTTPS?.....	206
3.1.10 Can Other Users Access My Data Stored in OBS?.....	206
3.1.11 Does OBS Support Resumable Transfer?.....	206
3.1.12 Does OBS Support Batch Upload?.....	206
3.1.13 Does OBS Support Batch Download?.....	207
3.1.14 Does OBS Support Batch Deletion of Objects?.....	207
3.1.15 What Are the Factors That Affect Upload and Download Speed of OBS?.....	208
3.1.16 Why Did Some of My Data Stored on OBS Get Lost?.....	208
3.1.17 Can Deleted Data Be Recovered?.....	208
3.1.18 Will There Be Data Left Over in OBS After I Delete an Object?.....	208
3.1.19 What Are the Differences Between OBS, EVS, and SFS?.....	208
3.1.20 Will My Bucket Performance Be Affected by Other Users' Services?.....	210
3.2 Access Control.....	210
3.2.1 How Can I Control Access to OBS?.....	210
3.2.2 What Are the Differences Between Using an IAM Policy and a Bucket Policy in Access Control?....	211
3.2.3 What Is the Relationship Between a Bucket Policy and an Object Policy?.....	211
3.2.4 Why Is the Message "Access denied" Still Appearing After OBS System Permissions Were Assigned by IAM?.....	211
3.2.5 Why Does Message "Access denied" Appear After I Was Granted the Read and Write Permissions for a Bucket?.....	212
3.2.6 Why Can't I Access OBS (403 AccessDenied) After Being Granted with the OBS Access Permission? .....	213

3.2.7 How Do I Control Access to Folders in an OBS Bucket?.....	216
3.3 Buckets and Objects.....	216
3.3.1 Why Am I Unable to Create a Bucket?.....	216
3.3.2 Why Am I Unable to Upload an Object?.....	216
3.3.3 Why Am I Unable to Download an Object?.....	217
3.3.4 Why Can't I Delete a Bucket?.....	217
3.3.5 What Is the Relationship Between Bucket Storage Classes and Object Storage Classes?.....	217
3.3.6 Can I Modify the Region of a Bucket?.....	217
3.3.7 Can I Edit Objects in OBS Online?.....	217
3.3.8 How Do I Obtain the Access Path to an Object?.....	218
3.3.9 Why Can't I Search for Certain Objects in My Bucket?.....	218
3.3.10 What Should I Do If an Error Message Is Displayed When I Use Internet Explorer to Access an Object URL That Contains Chinese Characters?.....	219
3.3.11 How Do I Batch Delete a Large Number of Objects from a Bucket or Empty a Bucket?.....	220
3.4 Tools.....	222
3.4.1 When Downloading a Folder Using obsutil, the Download Speed Slows After the Folder Download Progress Reaches 90%.....	222
3.4.2 With obsutil, Downloading a File Fails After the Download Progress Reaches 99%.....	223
3.4.3 How Do I Use the obsutil cp Command to Enable Incremental Upload, Download, or Replication?.....	223
3.5 APIs and SDKs.....	223
3.5.1 What Are the Differences Between PUT and POST Upload Methods?.....	223
3.5.2 Failure with OBS SDK in Uploading a File Greater than 5 GB.....	224
3.5.3 Why Don't the Signatures Match?.....	224
3.6 Security.....	226
3.6.1 How Is Data Security Ensured in OBS?.....	226
3.6.2 Does OBS Scan My Data for Other Purposes?.....	226
3.6.3 Can Engineers Export My Data from the Background of OBS?.....	226
3.6.4 How Does OBS Protect My Data from Being Stolen?.....	226
3.6.5 Can a Pair of AK and SK Be Replaced When It Is Being Used to Access OBS?.....	226
3.6.6 Can Multiple Users Share One Pair of AK and SK to Access OBS?.....	226
3.7 Durability and Availability.....	226
3.7.1 What Are the Differences Between Single-AZ and Multi-AZ Storage in OBS?.....	227
3.7.2 What Redundancy Storage Techniques Does OBS Use?.....	227
3.8 How Do I Use Fragment Management?.....	227
3.8.1 Why Are Fragments Generated?.....	227
3.8.2 How Do I Manage Fragments?.....	228
3.9 How Do I Use Versioning?.....	228
3.9.1 Can I Upload an Object to a Folder Where a Namesake Object Already Exists?.....	228
3.9.2 Can I Recover a Deleted Object?.....	228
3.10 How Do I Use Tags?.....	228
3.10.1 Can I Search for a Bucket by Tag?.....	228
3.10.2 What Can I Do with Tags?.....	228

3.11 Event Notification.....	229
3.11.1 Which Events Can Trigger Event Notifications?.....	229
3.12 How Do I Use Lifecycle Management?.....	229
3.12.1 What Are the Application Scenarios of Lifecycle Management?.....	229
3.13 How Do I Use Static Website Hosting?.....	230
3.13.1 Can OBS Host My Static Websites?.....	230
3.13.2 Which Types of Websites Can I Use OBS to Host?.....	230
3.13.3 How Do I Obtain the Static Website Hosting Address of a Bucket?.....	230
3.14 How Do I Manage Domain Names?.....	230
3.14.1 Why Is the Message "NoSuchBucket" Displayed When I Use a User-Defined Domain Name to Access a Bucket That Can Be Accessed by the OBS Domain Name?.....	231
3.14.2 What Is the Relationship Between OBS Bucket Names and Domain Names?.....	231
3.15 Monitoring.....	231
3.15.1 Why Can't I Find the Statistics on OBS 5XX Status Codes on Cloud Eye?.....	231
3.16 Server-Side Encryption.....	231
3.16.1 Does OBS Support Encrypted Upload?.....	231
3.16.2 What Encryption Technologies Can I Use to Encrypt Data on OBS?.....	232
3.16.3 Will OBS Server-Side Encryption Encrypt My Existing Objects That Are Unencrypted?.....	233
<b>A Change History.....</b>	<b>234</b>

# 1 Service Overview

---

## 1.1 About OBS

### OBS Overview

Object Storage Service (OBS) is a scalable service that provides secure, reliable, and cost-effective cloud storage for massive amounts of data.

OBS provides unlimited storage capacity for objects of any format, catering to the needs of common users, websites, enterprises, and developers. There is no limitation on the storage capacity of the entire OBS system or of a single bucket, and any number of objects can be stored. As a web service, OBS supports APIs over Hypertext Transfer Protocol (HTTP) and Hypertext Transfer Protocol Secure (HTTPS). You can use OBS Console or OBS tools to access and manage data stored in OBS anytime, anywhere. With OBS SDKs and APIs, you can easily manage data stored in OBS and develop upper-layer applications.

### Product Architecture

OBS basically consists of **buckets** and **objects**.

A bucket is a container for storing objects in OBS. Each bucket is specific to a region and has specific storage class and access permissions. A bucket is accessible through its **access domain name** over the Internet.

An object is the fundamental storage unit in OBS. An object consists of the following:

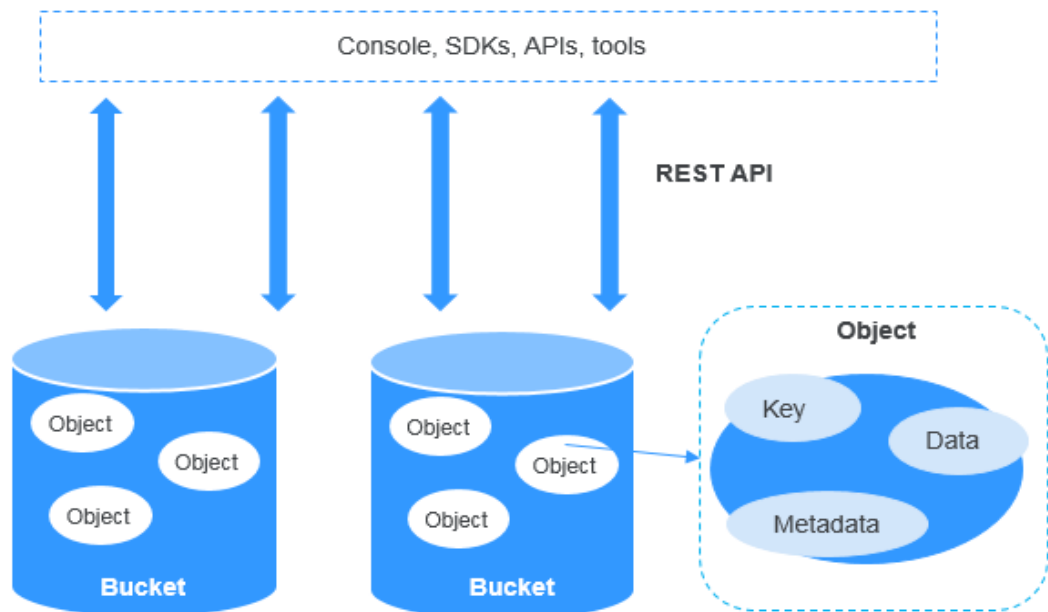
- A key that specifies the name of an object. An object key is a UTF-8 string up to 1,024 characters long. Each object is uniquely identified by a key within a bucket.
- Metadata that describes an object. The metadata is a set of key-value pairs that are assigned to objects stored in OBS. There are two types of metadata:
  - System-defined metadata is automatically assigned by OBS for processing objects. Such metadata includes Date, Content-Length, Last-Modified, ETag, and more.



- You can specify custom metadata to describe the object when you upload an object to OBS.
- Data that refers to the content of an object.

By means of secondary development based on OBS REST APIs, OBS Console, SDKs, and a variety of tools are provided for you to use OBS. You can also use OBS SDKs and APIs to develop applications customized for your business needs.

**Figure 1-1** Product architecture



## Storage Classes

OBS offers the storage classes below to meet your requirements for storage performance and cost:

- **Standard:** The Standard storage class features low latency and high throughput. It is therefore good for storing frequently (multiple times per month) accessed files or small files (less than 1 MB). Its application scenarios include big data analytics, mobile apps, hot videos, and social apps.
- **Warm:** The Warm storage class is for storing data that is infrequently (less than 12 times per year) accessed, but when needed, the access has to be fast. It can be used for file synchronization, file sharing, enterprise backups, and many other scenarios. This storage class has the same durability, low latency, and high throughput as the Standard storage class, with a lower cost, but its availability is slightly lower than the Standard storage class.
- **Cold:** The Cold storage class is ideal for storing data that is rarely (once per year) accessed. Its application scenarios include data archive and long-term backups. This storage class is secure, durable, and inexpensive, so it can be used to replace tape libraries. To keep cost low, it may take hours to restore data from the Cold storage class.

An object uploaded to a bucket inherits the storage class of the bucket by default. You can also specify a storage class for an object when you upload it.

Changing the storage class of a bucket does not change the storage classes of existing objects in the bucket, but newly uploaded objects will inherit the new storage class.

**Table 1-1** Comparison between storage classes

Compared Item	Standard	Warm	Cold
Feature	Top-notch performance, high reliability and availability	Reliable, inexpensive storage with real-time access	Long-term retention of archived data at a low cost
Application scenarios	Cloud application, data sharing, content sharing, and hot data storage	Web disk applications, enterprise backup, active archiving, and data monitoring	Archive, medical image storage, video material storage, and replacement of tape libraries
Minimum storage duration	N/A	30 days	90 days
Minimum measurement unit <sup>a</sup>	64 KB	64 KB	64 KB
<b>Data restore</b>	N/A	Billed for each GB restored.	Data can be restored at a standard, bulk, or an expedited speed. Billed for each GB restored.
Image processing	Supported	Supported	Not supported

## How to Access OBS

OBS provides various resource management tools. You can use any of the tools listed in [Table 1-2](#) to access and manage resources in OBS.

**Table 1-2** OBS resource management tools

Tool	Description
OBS Console	OBS Console is a web-based GUI for you to easily manage OBS resources.
OBS Browser+	OBS Browser+ is a Windows client that lets you easily manage OBS resources from your desktop.

Tool	Description
obsutil	obsutil is a command line tool for you to perform common configuration and management operations on OBS. If you are comfortable using the command line interface (CLI), obsutil is recommended for batch processing and automated tasks.
SDKs	OBS SDKs encapsulate the REST API provided by OBS to simplify development. You can call API functions provided by the OBS SDKs to enjoy OBS capabilities.
API	OBS offers the REST API for you to access it from web applications with ease. By making API calls, you can upload and download data anytime, anywhere, over the Internet.

## 1.2 Advantages

### Comparison Between OBS and On-Premises Storage Servers

In this information era, it becomes increasingly difficult for conventional on-premises storage servers to deal with the fast-growing data of enterprises. [Table 1-3](#) compares OBS with on-premises storage servers.

**Table 1-3** Comparison between OBS and on-premises storage servers

Item	OBS	On-Premises Storage Server
Storage capacity	OBS provides unlimited storage capacity. All services and storage nodes are deployed in distributed clusters. You can expand each node or cluster separately, and you never have to worry about running out of space.	Such servers provide confined storage space due to the limited capacity of the hardware devices they use. When the storage space is not sufficient, you need to buy extra disks for manual expansion.
Security	OBS uses HTTPS and SSL protocols and encrypts data during uploads. To keep data in transit and at rest safe, OBS uses access key IDs (AKs) and secret access keys (SKs) to authenticate user identities and adopts a range of approaches including IAM policies, bucket policies, access control lists (ACLs), and uniform resource locator (URL) validation.	The owner and users are exposed to security risks from cyber attacks, technical vulnerabilities, and accidental operations.

Item	OBS	On-Premises Storage Server
Costs	OBS is an out-of-the-box service that has no initial capital investment or time or labor costs and frees you from O&M.	The initial deployment of on-premises servers requires high investments and a long construction period, but it quickly lags behind as enterprise businesses change so fast. Additional expenditures are required to ensure security.

## OBS Advantages

- **Data durability and service continuity:** OBS supports access of hundreds of millions of users.
- **Multi-level protection and authorization management:** Measures, including versioning, server-side encryption, URL validation, virtual private cloud (VPC)-based network isolation, access log audit, and fine-grained access control are provided to keep data secure and trusted.
- **Highly concurrent access for hundreds of billions of objects:** With intelligent scheduling and response, optimized access paths, and technologies such as transmission acceleration, event notifications, and big data vertical optimization, you can store hundreds of billions of objects in OBS and still experience smooth concurrent access with ultra-high bandwidth and low latency.
- **Easy use and management:** OBS provides standard REST APIs to help you quickly move your workloads to cloud. Storage resources are linearly, infinitely scalable, without compromising performance. You do not have to plan storage capacity beforehand or worry about expansion or reduction.

## 1.3 Application Scenarios

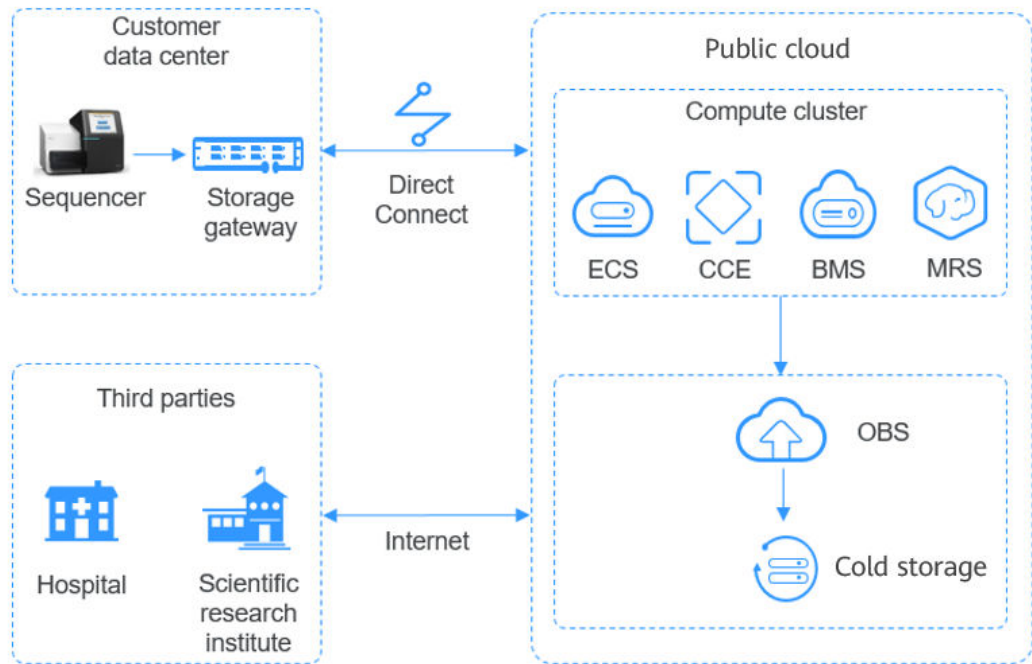
### DNA Sequencing

#### Scenario Description

OBS is a reliable, cost-effective system for storing massive amounts of data and features high concurrency and low latency. It works with compute services to help you easily build a DNA sequencing platform.

You can use Direct Connect to automatically upload data from the sequencer in your data center to the cloud. You can then perform data analysis on the compute cluster (including ECS, CCE, and MRS services), and the analysis results will be stored in OBS. After an analysis is completed, the source DNA data will be automatically stored in the Cold storage class in OBS, and the sequencing results can be distributed to hospitals and scientific research institutes over the Internet.

**Figure 1-2 DNA sequencing**



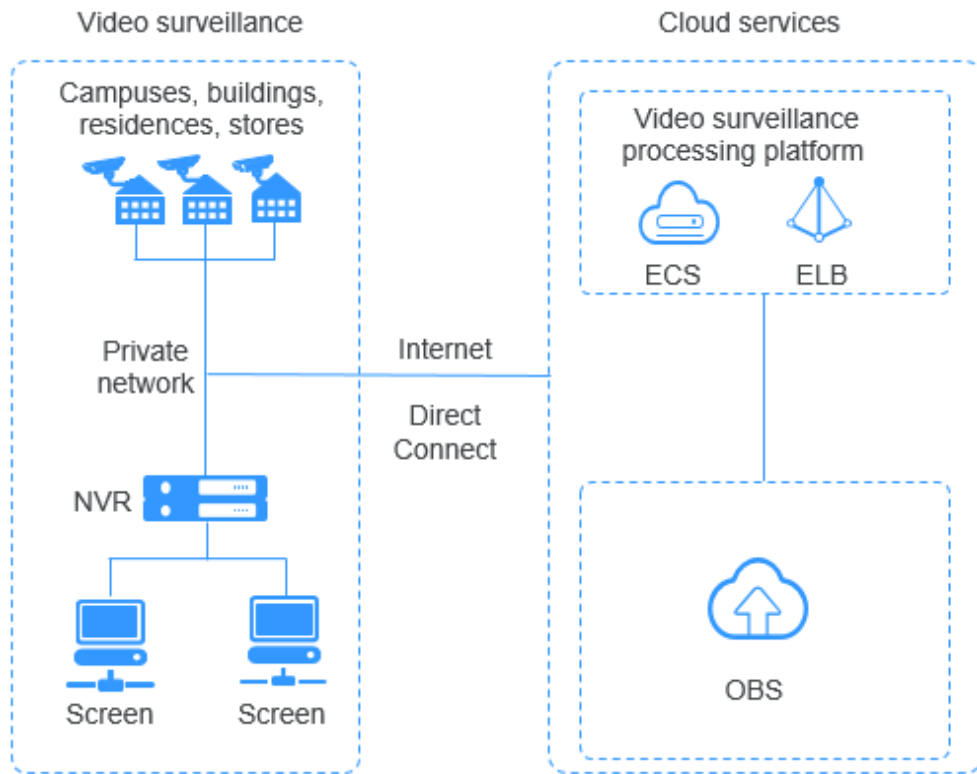
## Intelligent Video Surveillance

### Scenario Description

OBS provides reliable, inexpensive storage for virtually any amount of data. It features high performance and low latency and has a tiered storage class system (Standard, Warm, and Cold) to help reduce costs on storage.

You can upload surveillance videos recorded by cameras to the cloud over the Internet or using Direct Connect. Then segment the video files on the processing platform, which consists of ECS and ELB, and store video segmentation files in OBS. Later, you can download the video segmentation objects from OBS, and transfer them to terminal players.

**Figure 1-3** Video surveillance

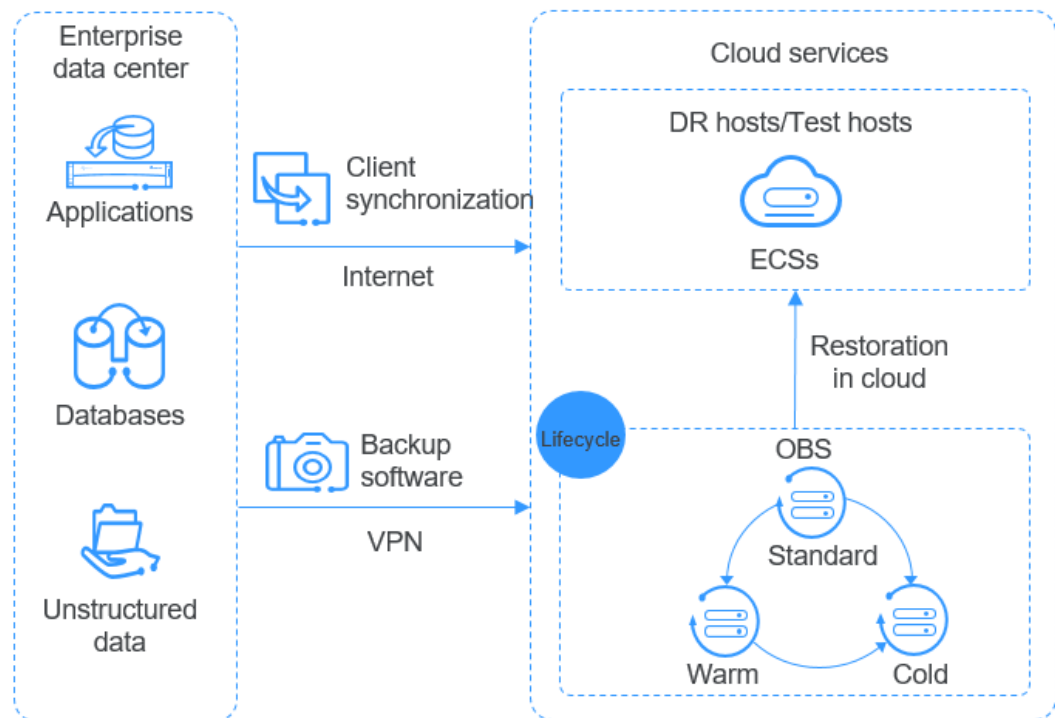


## Backup and Archiving

### Scenario Description

OBS offers a highly reliable, inexpensive storage system featuring high concurrency and low latency. It can hold massive amounts of data, meeting the archive needs for unstructured data of applications and databases.

**Figure 1-4 Backup and archiving**



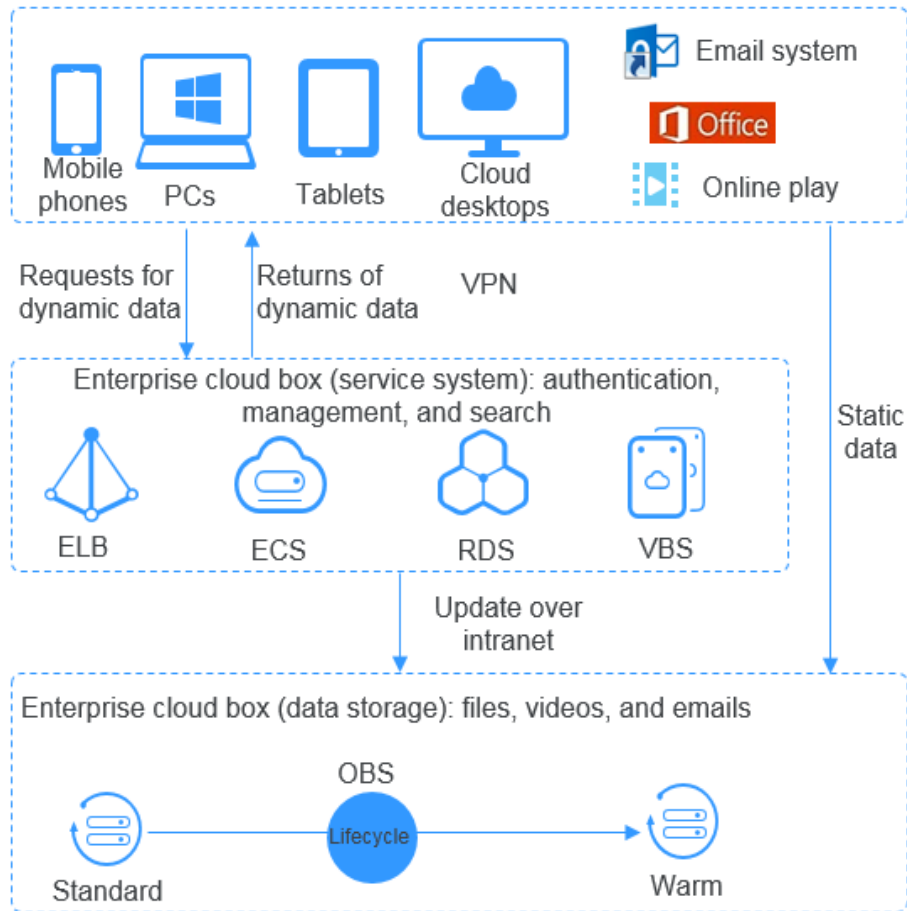
## Enterprise Cloud Boxes (Web Disks)

### Scenario Description

OBS works with cloud services such as ECS, ELB, RDS, and VBS to provide enterprise web disks with a reliable, inexpensive storage system featuring low latency and high concurrency. The storage capacity automatically scales as the volume of stored data grows.

Dynamic data on devices such as mobile phones, PCs, and tablets interacts with the enterprise cloud disk service system built on the cloud. Requests for dynamic data are sent to the service system for processing and then returned to devices, and the static data is stored in OBS. Service systems can process static data over the intranet. End users can directly request and read the static data from OBS. In addition, OBS provides the lifecycle management function to automatically change storage classes for objects, reducing storage costs.

**Figure 1-5** Enterprise cloud boxes (web disks)



## 1.4 Permissions Management

You can use Identity and Access Management (IAM) to manage OBS permissions and control access to your resources. IAM provides identity authentication, permissions management, and access control.

You can create IAM users for your employees, and assign permissions to these users on a principle of least privilege (PoLP) basis to control their access to specific resource types. For example, you can create IAM users for software developers and assign specific permissions to allow them to use OBS resources but prevent them from being able to delete resources or perform any high-risk operations.

If your account does not require individual IAM users for permissions management, skip this section.

IAM is a free service. You only pay for the resources in your account. For more information about IAM, see section "Service Overview" in the *Identity and Access Management User Guide*.

### OBS Permissions

By default, new IAM users do not have any permissions assigned. You can assign permissions to these users by adding them to one or more groups and attaching



policies to the groups. IAM provides preset system policies that define common permissions for different services, such as full control access and read-only. You can directly use these preset policies.

OBS is a global service deployed and accessed without specifying any physical region. OBS permissions are assigned to users in the global project, and users do not need to switch regions when accessing OBS.

#### Policy Types

- **RBAC policy:** An RBAC policy consists of permissions for an entire service. Users in a group with such a policy assigned are granted all the required permissions, including permissions for accessing and managing that service. RBAC policies do not support operation-specific permission control.
- **Fine-grained policy:** A fine-grained policy consists of API-based permissions for operations on specific resource types. Fine-grained policies, as the name suggests, allow for more fine-grained control than RBAC policies. Users with such fine-grained permissions can only perform specific operations on services.

#### NOTE

Due to data caching, an RBAC policy and fine-grained policy involving OBS actions will take effect 10 to 15 minutes after it is attached to a user, an enterprise project, and a user group.

**Table 1-4** lists all system policies of OBS.

**Table 1-4** OBS system policies

Policy	Description	Policy Type
Tenant Administrator	Allows you to perform any operation on all cloud resources under the account. OBS policies are configured under <b>Global service &gt; OBS</b> .	RBAC policy
Tenant Guest	Allows you to perform read-only operations on all cloud resources under the account. OBS policies are configured under <b>Global service &gt; OBS</b> .	RBAC policy
OBS Buckets Viewer	Allows you to list buckets, obtain basic bucket information and bucket metadata, and list objects. OBS policies are configured under <b>Global service &gt; OBS</b> .	RBAC policy

Policy	Description	Policy Type
OBS Viewer	<p>Allows you to list buckets, obtain basic bucket information and bucket metadata, and list objects.</p> <p>This policy is a system-defined policy of fine-grained authorization. Users with fine-grained authorization can use this policy and can create custom policy template based on this policy.</p> <p>OBS policies are configured under <b>Global service &gt; OBS</b>.</p>	Fine-grained policy
OBS Operator	<p>Allows you to perform all operations defined in OBS Viewer and to perform basic object operations, such as uploading objects, downloading objects, deleting objects, and obtaining object ACLs.</p> <p>This policy is a system-defined policy of fine-grained authorization. Users with fine-grained authorization can use this policy and can create custom policy template based on this policy.</p> <p>OBS policies are configured under <b>Global service &gt; OBS</b>.</p>	Fine-grained policy

The following table lists operations that can be performed under each set of OBS permission.

**Table 1-5** Permissions and the allowed operations on OBS resources

Operation	Tenant Administrator	Tenant Guest	OBS Buckets Viewer	OBS Viewer	OBS Operator
Listing buckets	Yes	Yes	Yes	Yes	Yes
Creating buckets	Yes	No	No	No	No
Deleting buckets	Yes	No	No	No	No
Obtaining basic bucket information	Yes	Yes	Yes	Yes	Yes

Operation	Tenant Administrator	Tenant Guest	OBS Buckets Viewer	OBS Viewer	OBS Operator
Controlling bucket access	Yes	No	No	No	No
Managing bucket policies	Yes	No	No	No	No
Modifying bucket storage classes	Yes	No	No	No	No
Listing objects	Yes	Yes	Yes	Yes	Yes
Listing objects with multiple versions	Yes	Yes	No	No	No
Uploading files	Yes	No	No	No	Yes
Creating folders	Yes	No	No	No	Yes
Deleting files	Yes	No	No	No	Yes
Deleting folders	Yes	No	No	No	Yes
Downloading files	Yes	Yes	No	No	Yes
Deleting files with multiple versions	Yes	No	No	No	Yes
Downloading files with multiple versions	Yes	Yes	No	No	Yes
Modifying object storage classes	Yes	No	No	No	No

<b>Operation</b>	<b>Tenant Administrator</b>	<b>Tenant Guest</b>	<b>OBS Buckets Viewer</b>	<b>OBS Viewer</b>	<b>OBS Operator</b>
Restoring files	Yes	No	No	No	No
Canceling the deletion of files	Yes	No	No	No	Yes
Deleting fragments	Yes	No	No	No	Yes
Controlling object access	Yes	No	No	No	No
Configuring object metadata	Yes	No	No	No	No
Obtaining object metadata	Yes	Yes	No	No	Yes
Managing versioning	Yes	No	No	No	No
Managing logging	Yes	No	No	No	No
Managing event notifications	Yes	No	No	No	No
Managing tags	Yes	No	No	No	No
Managing lifecycle rules	Yes	No	No	No	No
Managing static website hosting	Yes	No	No	No	No
Managing CORS rules	Yes	No	No	No	No
Managing URL validation	Yes	No	No	No	No

Operation	Tenant Administrator	Tenant Guest	OBS Buckets Viewer	OBS Viewer	OBS Operator
Managing domain names	Yes	No	No	No	No
Managing image processing	Yes	No	No	No	No
Appending objects	Yes	No	No	No	Yes
Configuring object ACL	Yes	No	No	No	No
Configuring the ACL for an object of a specified version	Yes	No	No	No	No
Obtaining object ACL information	Yes	Yes	No	No	Yes
Obtaining the ACL information of a specified object version	Yes	Yes	No	No	Yes
Uploading in the multipart mode	Yes	No	No	No	Yes
Listing uploaded parts	Yes	Yes	No	No	Yes
Canceling multipart uploads	Yes	No	No	No	Yes

## OBS Resource Permissions Management

Access to OBS buckets and objects can be controlled by IAM user permissions, bucket policies, and ACLs.

For more information, see [Overview](#).

## 1.5 Restrictions and Limitations

This section describes the restrictions on using OBS features.

**Table 1-6** OBS use restrictions and limitations

Item	Description
Bandwidth	By default, the maximum bandwidth for read/write (GET/PUT) requests of a single account is 16 Gbit/s. If the actual bandwidth reaches the threshold, flow control will be triggered.
Queries per second (QPS)	<p>Default maximum QPS allowed by a single account:</p> <ul style="list-style-type: none"> <li>• 6,000 write requests (PUT Object) per second</li> <li>• 10,000 read requests (GET Object) per second</li> <li>• 1,000 listing requests (LIST) per second</li> </ul> <p><b>NOTE</b> If you use sequential prefixes (such as timestamps or alphabetical order) for object naming, object access requests may be concentrated in a specific partition, resulting in access hotspots. This limits the request rate in a hotspot partition and increases access delay.</p> <p>Random prefixes are recommended for naming objects so that requests are evenly distributed across partitions, achieving horizontal expansion.</p>
Access rules	<p>In consideration of the DNS resolution performance and reliability, OBS requires that the bucket name must precede the domain when a request carrying a bucket name is constructed to form a three-level domain name, also mentioned as virtual-hosted-style access domain name.</p> <p>For example, you have a bucket named <b>test-bucket</b> in the <b>ru-moscow-1</b> region, and you want to access the ACL of an object named <b>test-object</b> in the bucket. The correct URL is <b>https://test-bucket.obs.ru-moscow-1.hc.sbercloud.ru/test-object?acl</b>.</p>

Item	Description
Buckets	<ul style="list-style-type: none"> <li>● On OBS, each bucket name must be unique and cannot be changed.</li> <li>● After you create a bucket, its name, storage redundancy policy, and region cannot be changed.</li> <li>● An account (including all IAM users under this account) can create a maximum of 100 buckets and parallel file systems. You can use the fine-grained access control of OBS to properly plan and use buckets.</li> <li>● By default, there is no limit on the storage capacity of the entire OBS system or a single bucket, and any number of objects can be stored.</li> <li>● A bucket can be deleted only after all objects in the bucket have been deleted.</li> <li>● The name of a deleted bucket can be reused for another bucket or a parallel file system at least 30 minutes after the deletion.</li> </ul>
Uploading objects	<ul style="list-style-type: none"> <li>● OBS Console supports uploading files in a batch. A maximum of 100 files can be uploaded in a batch with the total size of no more than 5 GB. If you upload only one file in a batch upload, it cannot exceed 5 GB in size.</li> <li>● If you use OBS Browser+, obsutil, an SDK, or an API, you can upload a single object of up to 48.8 TB.</li> <li>● If versioning is disabled for your bucket and you upload a new file with the same name as the one you previously uploaded to your bucket, the new file automatically overwrites the previous file and does not retain its ACL information. If you upload a new folder using the same name that was used with a previous folder in the bucket, the two folders will be merged, and files in the new folder will overwrite namesake files in the previous folder.</li> <li>● After versioning is enabled for your bucket, if the new file you upload has the same name as the one you previously uploaded to the bucket, a new file version will be added in the bucket.</li> <li>● Though any UTF-8 characters can be used in object keys (object names), it is recommended that object keys be named according to the <a href="#">object key naming guidelines</a>. These guidelines help object key names substantially meet the requirements of DNS, web security characters, XML analyzers, and other APIs.</li> </ul>
Deleting objects	<p>If versioning is not enabled for a bucket, deleted objects cannot be recovered.</p>

Item	Description
Restoring Cold objects	<ul style="list-style-type: none"> <li>● If a Cold object is being restored, you cannot suspend or delete the restore task.</li> <li>● You cannot restore an object in the <b>Restoring</b> state.</li> <li>● After an object is restored, an object copy in the Standard storage class will be generated. This way, there is a Cold object and a Standard object copy in the bucket at the same time. The Standard object copy will be automatically deleted upon its expiration.</li> </ul>
Lifecycle management	<p>There is no limit on the number of lifecycle rules in a bucket, but the total size of XML descriptions about all lifecycle rules in a bucket cannot exceed 20 KB.</p>
User-defined domain name binding	<ul style="list-style-type: none"> <li>● Only buckets in version 3.0 support user-defined domain name binding.</li> <li>● Currently, user domain names bound to OBS only allow access requests over HTTP.</li> <li>● A user-defined domain name can be bound to only one bucket.</li> <li>● Currently, the suffix of a user-defined domain name can contain 2 to 6 uppercase or lowercase letters.</li> </ul>
ACLs	<ul style="list-style-type: none"> <li>● A bucket ACL can have up to 100 grants. The total bucket ACL size cannot exceed 50 KB.</li> <li>● An object ACL can have up to 100 grants. The total object ACL size cannot exceed 50 KB.</li> </ul>
Bucket policies	<p>There is no limit on the number of bucket policies (statements) for a bucket, but the JSON descriptions of all bucket policies in a bucket cannot exceed 20 KB in total.</p>
Parallel file systems	<p>See the <i>Object Storage Service Parallel File System Feature Guide</i>.</p>
Image processing	<p>See the <i>Object Storage Service Image Processing Feature Guide</i>.</p>



## 1.6 Related Services

**Table 1-7** Related services

Function	Related Service	Reference
IAM provides the following functions: <ul style="list-style-type: none"> <li>• User identity authentication</li> <li>• IAM user permission control</li> <li>• IAM agency configuration</li> </ul>	Identity and Access Management (IAM)	<a href="#">Permissions Management</a> <a href="#">Configuring User Permissions</a>
Cloud Eye monitors OBS buckets, to collect statistics about the upload traffic, download traffic, the number of GET and PUT requests, the average Time to First Byte (TTFB) of GET requests, and the number of 4xx and 5xx errors.	Cloud Eye	<a href="#">OBS Monitoring Metrics</a>
CTS collects records of operations on OBS resources, facilitating querying, audits, and backtracking.	Cloud Trace Service (CTS)	<a href="#">Cloud Trace Service</a>
SMN sends OBS related alarms and event notifications, and triggers workflows.	Simple Message Notification (SMN)	<a href="#">SMN-Enabled Event Notifications</a>
Tags are used to label and classify buckets in OBS.	Tag Management Service (TMS)	<a href="#">Tag Overview</a>
KMS encrypts files uploaded to the OBS.	Data Encryption Workshop (DEW)	<a href="#">Server-Side Encryption Overview</a>

Function	Related Service	Reference
DNS resolves domain names configured for static website hosting in OBS.	Domain Name Service (DNS)	<a href="#">Using a User-Defined Domain Name to Configure Static Website Hosting</a>

OBS can serve as a storage resource pool for other cloud services such as Relational Database Service (RDS) and Cloud Trace Service (CTS).

## 1.7 Basic Concepts

### 1.7.1 Objects

Objects are basic units stored in OBS. An object contains both data and the metadata that describes data attributes. Data uploaded to OBS is stored in buckets as objects.

An object consists of the following:

- A key that specifies the name of an object. An object key is a UTF-8 string up to 1,024 characters long. Each object is uniquely identified by a key within a bucket.
- Metadata that describes an object. The metadata is a set of key-value pairs that are assigned to objects stored in OBS. There are two types of metadata: system-defined metadata and custom metadata.
  - System-defined metadata is automatically assigned by OBS for processing objects. Such metadata includes Date, Content-Length, Last-Modified, ETag, and more.
  - You can specify custom metadata to describe the object when you upload an object to OBS.
- Data that refers to the content of an object.

Generally, objects are managed as files. However, OBS is an object-based storage service and there is no concept of files and folders. For easy data management, OBS provides a method to simulate folders. By adding a slash (/) to an object name, for example, **test/123.jpg**, you can specify **test** as a folder and **123.jpg** as the name of a file in the **test** folder. The key of the object is **test/123.jpg**.

When uploading an object, you can set a storage class for the object. If no storage class is specified, the object is stored in the same storage class as the bucket in which it resides. You can also change the storage class of an existing object in a bucket.

On OBS Console, you can use folders the same way you use them in a file system.

## Object Key Naming Guidelines

Although any UTF-8 characters can be used in an object key name, naming object keys according to the following guidelines can help maximize the object keys' compatibility with other applications. Ways to analyze special characters vary depending on applications. The following guidelines help object key names substantially meet the requirements of DNS, web security characters, XML analyzers and other APIs.

The following character sets can be safely used in key names.

Alphanumeric characters (also known as unreserved characters)	0-9, a-z, and A-Z
Special characters (also known as reserved characters)	Exclamation mark (!) Hyphen (-) Underscore (_) Period (.) Asterisk (*) Single quote (') Left parenthesis ( Right parenthesis (>)

The following are examples of valid object key names:

```
4my-organization
my.great_photos-2014/jan/myvacation.jpg
videos/2014/birthday/video1.wmv
```

### 1.7.2 Buckets

Buckets are containers for storing objects. OBS provides flat storage in the form of buckets and objects. Unlike the conventional multi-layer directory structure of file systems, all objects in a bucket are stored at the same logical layer.

Each bucket has its own attributes, such as access permissions, storage class, and the region. You can specify access permissions, storage class, and regions when creating buckets. You can also configure advanced attributes to meet storage requirements in different scenarios.

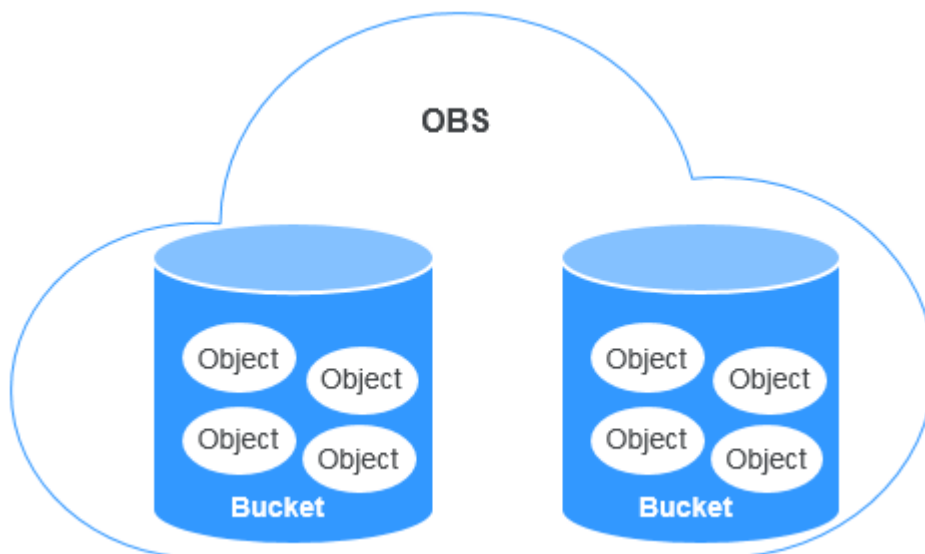
Each bucket name in OBS is globally unique and cannot be changed after the bucket has been created. The region where a bucket resides cannot be changed once the bucket is created. When you create a bucket, OBS creates a default access control list (ACL) that grants users permissions (such as read and write permissions) on the bucket. Only authorized users can perform operations such as creating, deleting, viewing, and configuring buckets.

An account (including all IAM users under this account) can create a maximum of 100 buckets and parallel file systems. However, there is no restriction on the number and total size of objects in a bucket.

OBS adopts the REST architectural style, and is based on HTTP and HTTPS. You can use URLs to locate resources.

**Figure 1-6** illustrates the relationship between buckets and objects in OBS.

**Figure 1-6** Relationship between objects and buckets



### 1.7.3 Parallel File System

Parallel File System (PFS) is a high-performance semantic file system provided by OBS. It features access latency in milliseconds, TB/s-level bandwidth, and millions of IOPS, which makes it ideal for processing high-performance computing (HPC) workloads.

For details about PFS, see the *Parallel File System Feature Guide*.

### 1.7.4 Access Keys (AK/SK)

OBS uses an access key ID (AK) and secret access key (SK) to authenticate the identity of a requester. When you use OBS APIs for secondary development and use the AK and SK for authentication, the signature must be calculated based on the algorithm defined by OBS and added to the request.

The authentication can be based on a permanent AK and SK pair, or based on a temporary AK/SK pair and security token.

#### Permanent AK/SK Pair

You can create a pair of permanent AK and SK on the **My Credentials** page.

- Access key ID (AK): indicates the ID of the access key. It is the unique ID associated with the SK. The AK and SK are used together to obtain an encrypted signature for a request.
- Secret access key (SK): indicates the private key used together with its associated AK to cryptographically sign requests. The AK and SK are used together to identify a request sender to prevent the request from being modified.

#### Temporary AK/SK Pair

A temporary AK/SK pair and security token assigned by OBS comply with the principle of least privilege and are for temporarily accessing OBS. They are valid from 15 minutes to 24 hours, and need to be obtained again once they expire. If the security token is missing from your request, a 403 error will be returned.

- **Temporary AK:** indicates the ID of a temporary access key. It is the unique ID associated with the SK. The AK and SK are used together to obtain an encrypted signature for a request.
- **Temporary SK:** indicates the temporary private key used together with its associated temporary AK. The AK and SK are used together to identify a request sender to prevent the request from being modified.
- **Security token:** indicates the token used together with the temporary AK and SK to access all resources of a specified account.

When using the following tools to access OBS resources, you need to use the AK/SK pair for security authentication.

**Table 1-8** OBS resource management tools

Tool	AK/SK Configuration
OBS Browser+	Configure the AK and SK during account configuration.
obsutil	Configure the AK and SK during initial configuration.
obsfs	Configure the AK and SK during initial configuration.
SDKs	Configure the AK and SK in the initialization phase.
APIs	Add the AK/SK pair to the request when computing the signature.

## 1.7.5 Endpoints and Domain Names

**Endpoint:** OBS provides an endpoint for each region. An endpoint is considered a domain name to access OBS in a region and is used to process requests of that region. For details about regions and endpoints, see [Regions and Endpoints](#).

**Bucket domain name:** Each bucket in OBS has a domain name. A domain name is the address of a bucket and can be used to access the bucket over the Internet. It is applicable to cloud application development and data sharing.

An OBS bucket domain name is in the format of *BucketName.Endpoint*, where *BucketName* indicates the name of the bucket, and *Endpoint* indicates the domain name of the region where the bucket is located.

[Table 1-9](#) lists the bucket domain name and other domain names in OBS, including their structure and protocols.

**Table 1-9** OBS domain names

Type	Structure	Description	Protocol
Regional domain name	<b>Endpoint</b>	Each region has an endpoint, which is the domain name of the region.  For regions and endpoints, see <a href="#">Regions and Endpoints</a> .	HTTPS HTTP
Bucket domain name	<b>BucketName.Endpoint</b>	After a bucket is created, you can use the domain name to access the bucket. You can compose the domain name according to the structure of bucket domain names, or you can obtain it from basic information of the bucket on OBS Console or OBS Browser+.	HTTPS HTTP
Object domain name	<b>BucketName.Endpoint/ObjectName</b>	After an object is uploaded to a bucket, you can use the object domain name to access the object. You can spell out the domain name according to the structure of object domain names, or you can obtain it from the object details on OBS Console or OBS Browser+.	HTTPS HTTP
Static website domain name	<b>BucketName.obs-website.Endpoint</b>	A static website domain name is a bucket domain name when the bucket is configured to host a static website.	HTTPS HTTP
User-defined domain name	Self-owned domain name registered with a domain name provider	You can bind a user domain name to a bucket so that you can access the bucket through the user domain name.	HTTP

## 1.7.6 Region and AZ

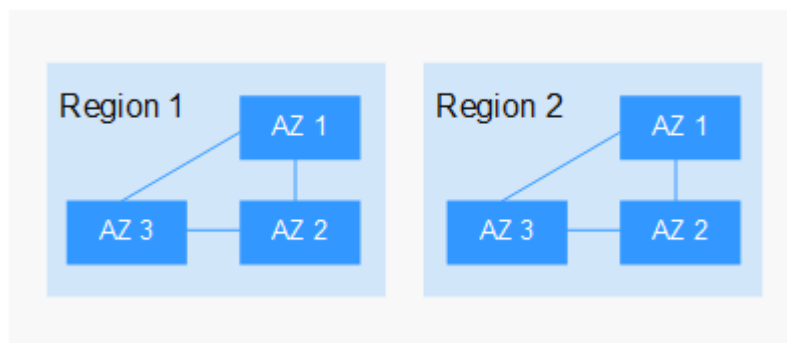
### Concept

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

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

[Figure 1-7](#) shows the relationship between the regions and AZs.

**Figure 1-7** Regions and AZs



### How Do I Select a Region?

You are advised to select a region close to you or your target users. This reduces network latency and improves access speed.

### How Do I Select an AZ?

When determining whether to deploy resources in the same AZ, consider your applications' requirements for disaster recovery (DR) and network latency.

- For high DR capability, deploy resources in different AZs in the same region.
- For low network latency, deploy resources in the same AZ.

### Regions and Endpoints

Before using an API to call resources, you must specify its region and endpoint. For details about public cloud regions and endpoints, see [Regions and Endpoints](#).

# 2 OBS Console Operation Guide

## 2.1 Console Function Overview

[Table 2-1](#) lists functions provided by OBS Console.

**Table 2-1** OBS Console functions

Function	Description
<a href="#">Basic bucket operations</a>	Allow you to create and delete buckets of different storage classes in specified regions (service areas), as well as change bucket storage classes.
<a href="#">Basic object operations</a>	Allow you to manage objects, including uploading (multipart uploads included), downloading, and deleting objects, as well as changing object storage classes and restoring Cold objects.
<a href="#">Server-side encryption</a>	Encrypts objects on the server side to enhance security of objects stored on OBS.
<a href="#">Object metadata</a>	Allows you to set properties for objects.
<a href="#">Monitoring</a>	<ul style="list-style-type: none"><li>• Cloud Eye can monitor the following OBS metrics:<ul style="list-style-type: none"><li>- Download Traffic</li><li>- Upload Traffic</li><li>- GET Requests</li><li>- PUT Requests</li><li>- First Byte Download Delay</li><li>- 4xx Errors</li><li>- 5xx Errors</li></ul></li></ul>
<a href="#">Auditing</a>	With Cloud Trace Service (CTS), you can record data operations associated with OBS for later query, audit, and backtrack operations.



Function	Description
<b>Fragment management</b>	Manages and clears fragments generated due to object upload failures.
<b>Versioning</b>	Stores multiple versions of an object in the same bucket.
<b>Logging</b>	Logs bucket access requests for analysis and auditing.
<b>Event notification</b>	Allows you to receive messages and emails from OBS.
<b>Permission control</b>	Controls access to OBS using IAM policies, bucket/object policies, and bucket/object access control lists (ACLs).
<b>Lifecycle management</b>	Allows you to configure lifecycle rules to periodically expire and delete objects or transition objects between storage classes.
<b>Tags</b>	Help you identify and classify buckets in OBS.
<b>Static website hosting</b>	Supports the hosting of static websites in buckets and the redirection of access requests for buckets.
<b>User-defined domain name configuration</b>	Enables you to bind your website domain name to a bucket domain name. If you want to migrate files from your website to OBS while keeping the website address unchanged, you can use this function.
<b>URL validation</b>	Prevents object links in OBS from being stolen by other websites.
<b>Cross origin resource sharing (CORS)</b>	Allows a web client in one origin to interact with resources in another one. Cross origin resource sharing (CORS) is a browser-standard mechanism defined by the World Wide Web Consortium (W3C). For general web page requests, website scripts and contents in one origin cannot interact with those in another because of Same Origin Policies (SOPs).

## 2.2 Restrictions

**Table 2-2** lists the web browser versions compatible with OBS Console.

**Table 2-2** Supported web browser versions

Web Browser	Version
Internet Explorer	<ul style="list-style-type: none"><li>• Internet Explorer 9 (IE9)</li><li>• Internet Explorer 10 (IE10)</li><li>• Internet Explorer 11 (IE11)</li></ul>
Firefox	Firefox 55 and later
Chrome	Chrome 60 and later

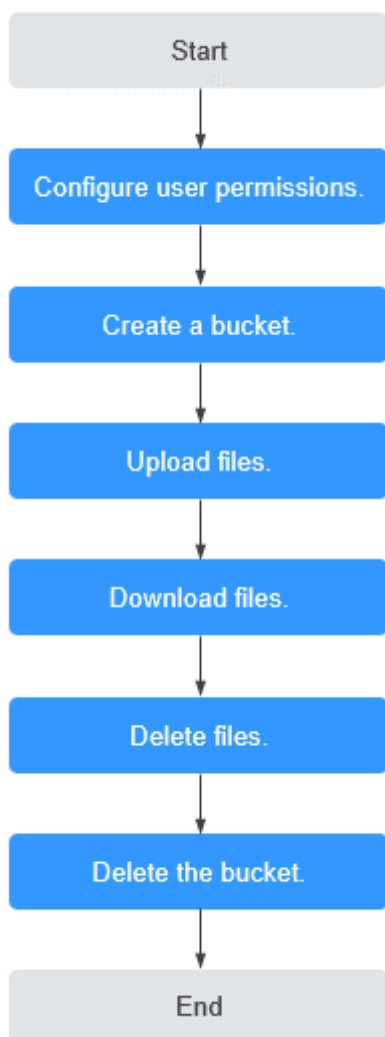
## 2.3 Getting Started

### 2.3.1 Process Description

OBS basic operations include bucket creation, object upload and object download.

The follow-up sections describe how to complete the tasks illustrated in [Figure 2-1](#).

**Figure 2-1** OBS Console quick start



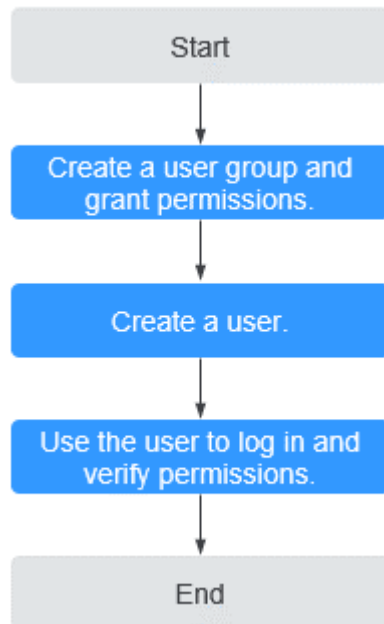
### 2.3.2 Configuring User Permissions

If your cloud service account does not need individual IAM users, then you may skip this section. Your permissions to use OBS functions are not affected.

If IAM users are required, you need to grant them access permissions for OBS, because OBS is separately deployed from other cloud resources.

## Process

**Figure 2-2** Process of granting an IAM user the OBS permissions



## Procedure

- Step 1** Log in to the management console with your account.
- Step 2** On the top menu bar, choose **Service List > Management & Deployment > Identity and Access Management**. The IAM console is displayed.
- Step 3** Create a user group and assign OBS permissions to it.

A user group is a collection of users. By assigning permissions to a user group, you assign permissions to the users in this group. After you create an IAM user, add it to one or more user groups, so that it can inherit the permissions from the groups.

1. In the navigation pane, choose **User Groups**. The **User Groups** page is displayed.
2. Click **Create User Group**.
3. Enter a user group name and click **OK**.  
The user group is displayed in the user group list once the creation is complete.
4. Locate the user group you created and click **Authorize** in the **Operation** column of the row.
5. Under **Select Policy/Role**, filter policies based on policy types in the upper right corner, required policy names, and click **Next**.
6. Under **Select Scope**, select **Global services** and click **OK**.

**NOTE**

In the policy content area, you can view the authorization details.

Due to data caching, an RBAC policy or a fine-grained policy involving OBS actions will take effect 10 to 15 minutes after it is attached to a user, an enterprise project, or a user group.

**Step 4** Create a user. For details, see [Creating an IAM user](#).

**Step 5** Use the created IAM user to log in to OBS Console and verify the user permissions.

----End

## 2.3.3 Creating a Bucket

This section describes how to create a bucket on OBS Console. A bucket is a container that stores objects in OBS. Before you can store data in OBS, you need to create a bucket.

**NOTE**

An account can create a maximum of 100 buckets and parallel file systems.

### Procedure

**Step 1** In the upper right corner of the OBS Console homepage, click **Create Bucket**. The **Create Bucket** page is displayed. For details, see [Figure 2-3](#).

**Figure 2-3** Creating a bucket

**Create Bucket**

Region: r

Regions are geographic areas isolated from each other. Resources are region-specific and cannot be used across regions through internal network connections. For low network latency and quick resource access, select the nearest region. Once a bucket is created, the region cannot be changed.

Bucket Name: Enter a bucket name. [View naming rules](#)

Cannot be the same as that of the current user's existing buckets. Cannot be the same as that of any other user's existing buckets. Cannot be edited after creation.

Data Redundancy Policy: Multi-AZ storage (selected) | Single-AZ storage

This setting can't be changed after the bucket is created. Multi-AZ storage is more expensive, but offers a higher availability. Data is stored in multiple AZs in the same region, improving availability.

Default Storage Class: Standard (selected) | Warm | Cold

Standard: For frequently accessed data. Warm: Less expensive, for infrequently accessed data. Cold: For data accessed once a year.

If you do not specify a storage class during object upload, any objects you upload inherit this default storage class.

Bucket Policies: Private (selected) | Public Read | Public Read/Write

Only the bucket owner has full control over the bucket.

Server-Side Encryption: SSE-KMS (selected) | Disable

If server-side encryption is enabled, new objects uploaded to this bucket are automatically encrypted. After a bucket is created, you can also change the encryption configuration on the overview page. Encryption is recommended to keep data secure.

Enterprise Project: --Select-- [Create Enterprise Project](#)

Tags: It is recommended that you use TMS's predefined tag function to add the same tag to different cloud resources. [View predefined tags](#)

Tag key: Tag value

**Create Now**

**Step 2** Configure bucket parameters.**Table 2-3** Bucket parameters

Parameter	Description
Region	Geographic area where a bucket resides. For low latency and faster access, select the region nearest to you. Once the bucket is created, its region cannot be changed.
Bucket Name	<p>Name of the bucket. A bucket name must be unique across all accounts and regions. Once a bucket is created, its name cannot be changed.</p> <p>According to the globally applied DNS naming rules, an OBS bucket name:</p> <ul style="list-style-type: none"><li>• Must be unique across all accounts and regions. The name of a deleted bucket can be reused for another bucket or a parallel file system at least 30 minutes later after the deletion.</li><li>• Must be 3 to 63 characters long. Only lowercase letters, digits, hyphens (-), and periods (.) are allowed.</li><li>• Cannot start or end with a period (.) or hyphen (-), and cannot contain two consecutive periods (..) or contain a period (.) and a hyphen (-) adjacent to each other.</li><li>• Cannot be formatted as an IP address.</li></ul> <p><b>NOTE</b> When you access OBS through HTTPS using virtual hosted-style URLs, if the bucket name contains a period (.), the certificate verification will fail. To work around this issue, you are advised not to use periods (.) in bucket names.</p>
Data Redundancy Policy	<ul style="list-style-type: none"><li>• <b>Multi-AZ storage:</b> Data is stored in multiple AZs to achieve higher reliability.</li><li>• <b>Single-AZ storage:</b> Data is stored in a single AZ, with lower costs.</li></ul> <p>Once a bucket is created, the data redundancy policy cannot be changed, so choose the policy that can meet your needs.</p> <ul style="list-style-type: none"><li>• Multi-AZ storage is not available for buckets in the Cold storage class.</li></ul>

Parameter	Description
Default Storage Class	<p>Storage classes of a bucket. Different storage classes meet different requirements for storage performance and costs.</p> <ul style="list-style-type: none"> <li>• The Standard storage class is for storing a large number of hot files or small files that are frequently accessed (multiple times per month on average) and require quick retrieval.</li> <li>• The Warm storage class is for storing data that is less frequently accessed (less than 12 times per year on average) and requires quick retrieval.</li> <li>• The Cold storage class is for archiving data that is rarely accessed (once a year on average) and has no requirements for quick retrieval.</li> </ul> <p>For details, see <a href="#">Storage Classes Overview</a>.</p>
Bucket Policy	<p>Controls read and write permissions for buckets.</p> <ul style="list-style-type: none"> <li>• <b>Private:</b> No access beyond the bucket ACL settings is granted.</li> <li>• <b>Public Read:</b> Anyone can read objects in the bucket.</li> <li>• <b>Public Read and Write:</b> Anyone can read, write, or delete objects in the bucket.</li> </ul>
Server-Side Encryption	<p>Select <b>SSE-KMS</b>. For the encryption key type, you can choose <b>Default</b> or <b>Custom</b>. If <b>Default</b> is used, the default key of the current region will be used to encrypt your objects. If there is no such a default key, OBS creates one the first time you upload an object. If <b>Custom</b> is used, you can choose a custom key you created on the KMS console and choose the project where the key belongs to encrypt your objects.</p> <p>When server-side encryption is enabled for a bucket, you can configure an object to inherit the bucket's KMS encryption settings when you upload the object to the bucket.</p>
Enterprise Project	<p>You can add a bucket to an enterprise project for unified management.</p> <p>Create an enterprise project on the enterprise project page. The default enterprise project is named <b>default</b>.</p> <p>On the <b>Enterprise Project Management</b> page, create an enterprise project, create a user group and add users to this group, and then add the user group to the enterprise project. By doing so, users in this user group obtain the operation permissions for the buckets and objects in the enterprise project.</p> <p><b>NOTE</b> Only an enterprise account can configure enterprise projects. OBS Viewer and OBS Operator are the fine-grained authorizations of the enterprise project user group in OBS.</p>

Parameter	Description
Tags	Optional. Tags are used to identify and classify buckets in OBS. Each tag is represented by a key-value pair. For more information, see <a href="#">Tag Overview</a> .

**Step 3** Click **Create Now**.

----End

## 2.3.4 Uploading an Object

This section describes how to upload local files to OBS over the Internet. These files can be texts, images, videos, or any other type of files.

### NOTE

OBS Console allows you to upload files in a batch. Up to 100 files can be uploaded at a time, with the total size of no more than 5 GB. If the file size exceeds 5 GB, but no larger than 48.8 TB, use tools (such as OBS Browser+ and obsutil) or the multipart upload of OBS SDKs and APIs for upload.

If versioning is disabled for your bucket and you upload a new file with the same name as the one you previously uploaded to your bucket, the new file automatically overwrites the previous file and does not retain its ACL information. If you upload a new folder using the same name that was used with a previous folder in the bucket, the two folders will be merged, and files in the new folder will overwrite namesake files in the previous folder.

After versioning is enabled for your bucket, if the new file you upload has the same name as the one you previously uploaded to the bucket, a new file version will be added in the bucket. For details about versioning, see [Versioning Overview](#).

## Prerequisites

- At least one bucket has been created.
- If you want to classify files, you can create folders and upload files to different folders. For details, see [Creating a Folder](#).

## Procedure

**Step 1** In the bucket list, click the bucket you want to operate to go to the **Objects** page.

**Step 2** Go to the folder where you want to upload files and click **Upload Object**. The **Upload Object** dialog box is displayed.

### NOTE

If the files that you want to upload to OBS are stored in Microsoft OneDrive, it is recommended that the names of these files contain a maximum of 32 characters to ensure compatibility.



Figure 2-4 Uploading objects

**Upload Object** [How to Upload a File Larger than 5 GB?](#) ×

1 Upload Object — (2) (Optional) Configure Advanced Settings

**i** Upload actions will generate requests. After the upload, you will be billed for data storage. ×

Object Permission: **Private** Public Read Public Read/Write

Storage Class: **Standard** Warm Cold

Optimized for frequently accessed (multiple times per month) data such as small and essential files that require low latency. If you do not change this setting, your uploaded objects will be stored using the default storage class you selected during bucket creation. [Learn more](#)

Upload Object: **i** The file or folder you newly upload will overwrite any existing file or folder with the same name. To keep different versions of the same file or folder, enable versioning for the current bucket.

Drag and drop files or folders, or [add files](#)  
(A maximum of 100 files can be uploaded at a time. The total size cannot exceed 5 GB.)

Next: (Optional) Configure Advanced Settings Upload Cancel

**Step 3** Specify the read and write permissions for the object.

**Step 4** Select a storage class. If you do not specify a storage class, the objects you upload inherit the default storage class of the bucket.

**NOTE**

An object can have a different storage class from its bucket. You can specify a storage class for an object when uploading it, or you can change the object storage class after the object is uploaded.

**Step 5** In the **Upload Object** area, drag and drop the files or folders you want to upload.

You can also click **add files** in the **Upload Object** area to select files.

**Step 6 Server-Side Encryption:** Choose **SSE-KMS** or **Disable**. For details, see [Uploading an Object in Server-Side Encryption Mode](#).

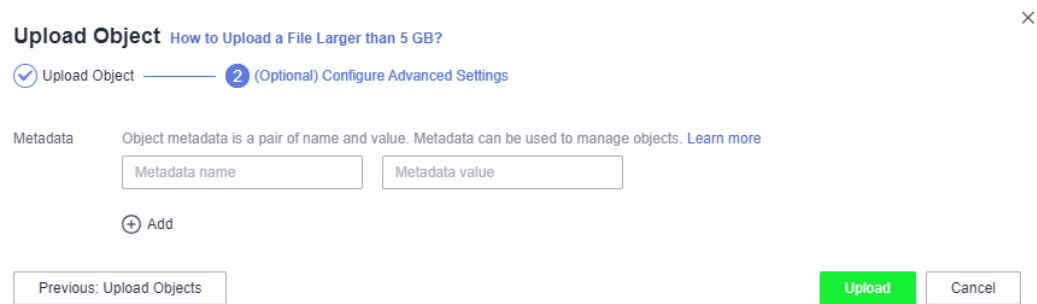
**NOTE**

If a bucket has server-side encryption configured, you can select **Inherit from bucket** when uploading an object to the bucket, for the object to inherit the encryption settings from the bucket.

**Step 7** (Optional) To configure metadata, click **Next: (Optional) Configure Advanced Settings**.

Add metadata ContentDisposition, ContentLanguage, WebsiteRedirectLocation, ContentEncoding, or ContentType as needed. For more information, see [Object Metadata](#). Metadata is a set of name-value pairs. The metadata value cannot be left blank. You can add two or more metadata entries by clicking **Add**.

**Figure 2-5** Configuring metadata



**Step 8** Click **Upload**.

----End

## 2.3.5 Downloading an Object

You can download files from OBS Console to your local computer.

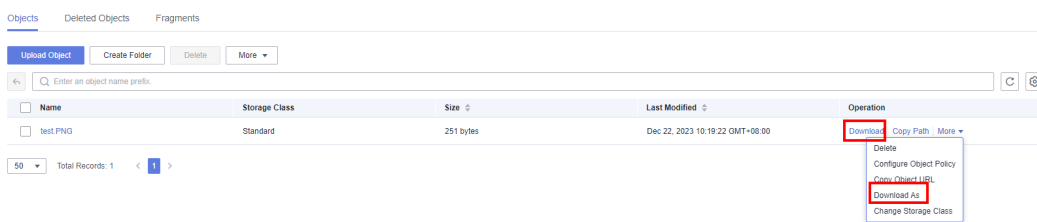
### Limitations and Constraints

Objects in the Cold storage class can be downloaded only when they are in the **Restored** state.

### Procedure

- Step 1** In the bucket list, click the bucket you want to operate to go to the **Objects** page.
- Step 2** Select the file you want to download, and click **Download** or choose **More > Download As** on the right.

**Figure 2-6** Downloading an object



#### NOTE

In the **Download As** dialog box, right-click the object and choose **Copy Link Address** from the shortcut menu to obtain the object's download address.

----End

## 2.3.6 Deleting an Object

You can delete unnecessary files one by one or in a batch on OBS Console to save space and money.

## Procedure

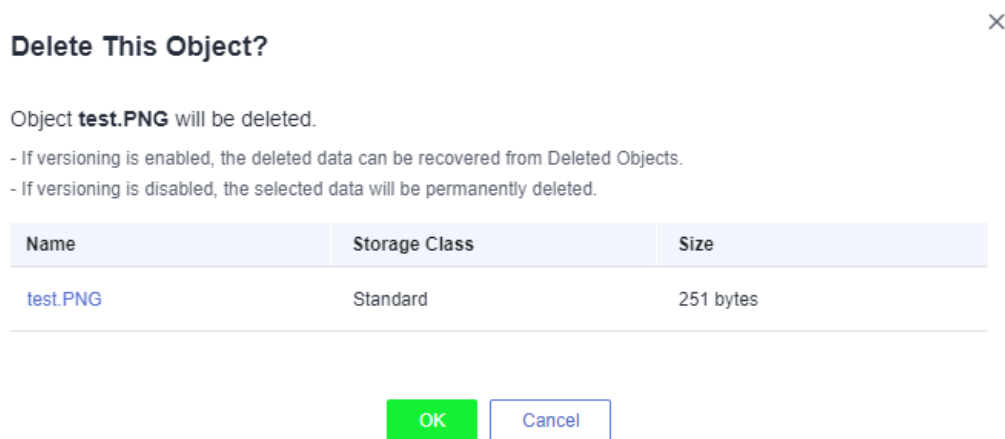
**Step 1** In the bucket list, click the bucket you want to operate to go to the **Objects** page.

**Step 2** Select the file you want to delete, and choose **More > Delete** on the right.

You can select multiple files and click **Delete** above the file list to batch delete them.

**Step 3** Click **OK** to confirm the deletion.

**Figure 2-7** Deleting an object



----End

## Important Notes

In big data scenarios, parallel file systems usually have deep directory levels and each directory has a large number of files. In such case, deleting directories from parallel file systems may fail due to timeout. To address this problem, you are advised to configure [a lifecycle rule](#) for directories so that they can be deleted in background based on the preset lifecycle rule.

### 2.3.7 Deleting a Bucket

You can delete unwanted buckets on OBS Console to free up the quota of buckets.

#### Prerequisites

- All objects in the bucket have been permanently deleted. A bucket must be emptied before it can be deleted.

---

#### NOTICE

Objects under the **Objects**, **Deleted Objects**, and **Fragments** tabs must be all deleted.

- A bucket can only be deleted by the bucket owner.

## Procedure

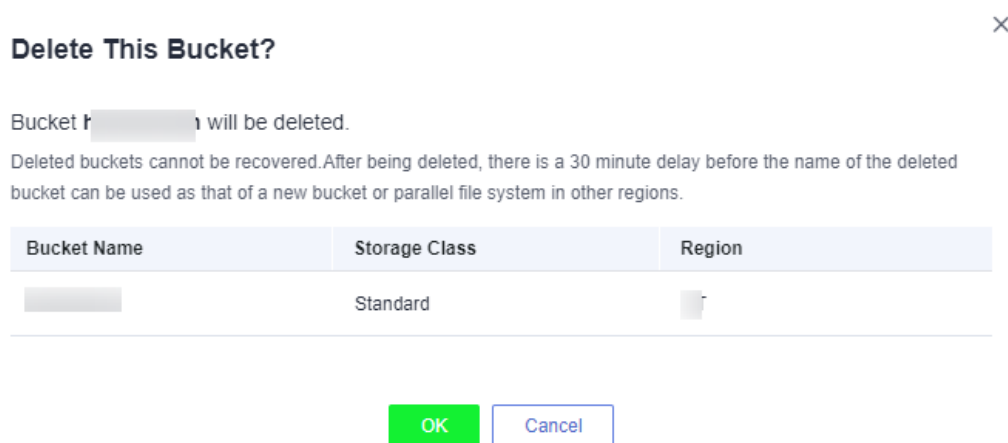
**Step 1** In the bucket list on OBS Console, select the bucket you want to delete, and then click **Delete** on the right.

 **NOTE**

The name of a deleted bucket can be reused for another bucket or parallel file system at least 30 minutes after the deletion.

**Step 2** Click **OK** to confirm the deletion.

**Figure 2-8** Deleting a bucket



----End

## 2.4 Storage Classes Overview

OBS supports tiered storage classes at the bucket level and object level.

OBS provides the following storage classes: Standard, Warm, and Cold.

These storage classes can meet different needs for storage performance and costs.

- **Standard:** The Standard storage class features low latency and high throughput. It is therefore good for storing frequently (multiple times per month) accessed files or small files (less than 1 MB). Its application scenarios include big data analytics, mobile apps, hot videos, and social apps.
- **Warm:** The Warm storage class is for storing data that is infrequently (less than 12 times per year) accessed, but when needed, the access has to be fast. It can be used for file synchronization, file sharing, enterprise backups, and many other scenarios.
- **Cold:** The Cold storage class is ideal for storing data that is rarely (once per year) accessed. Its application scenarios include data archive and long-term backups. The Cold storage class is secure, durable, and inexpensive, and can be used to replace tape libraries. To keep cost low, it may take hours to restore data from the Cold storage class.

## Bucket Storage Classes vs. Object Storage Classes

When an object is uploaded, it inherits the storage class of the bucket by default, but you can change the default storage class when you upload the object.

Changing the storage class of a bucket does not change the storage classes of existing objects in the bucket, but newly uploaded objects will inherit the new storage class.

### Comparison of Storage Classes

Compared Item	Standard	Warm	Cold
Feature	Top-notch performance, high reliability and availability	Reliable, inexpensive storage with real-time access	Long-term storage for Cold data at a low cost
Application scenarios	Cloud application, data sharing, content sharing, and hot data storage	Web disk applications, enterprise backup, active archiving, and data monitoring	Archive, medical image storage, video material storage, and replacement of tape libraries
Minimum measurement unit <sup>a</sup>	64 KB	64 KB	64 KB
Minimum storage duration	N/A	30 days	90 days
Data restore	N/A	Billed for each GB restored.	Data can be restored at a standard, bulk, or an expedited speed. Billed for each GB restored.
Image processing	Supported	Supported	Not supported

## 2.5 Managing Buckets

### 2.5.1 Creating a Bucket

A bucket is a container that stores objects in OBS. Before you store data in OBS, you need to create a bucket.

 **NOTE**

An account can create a maximum of 100 buckets and parallel file systems.

## Procedure

- Step 1** In the upper right corner of the OBS Console homepage, click **Create Bucket**. The **Create Bucket** page is displayed. For details, see [Figure 2-9](#).

**Figure 2-9** Creating a bucket

The screenshot shows the 'Create Bucket' configuration page in the OBS console. It includes the following sections:

- Region:** A dropdown menu with a selected region 'r'. A note below states: "Regions are geographic areas isolated from each other. Resources are region-specific and cannot be used across regions through internal network connections. For low network latency and quick resource access, select the nearest region. Once a bucket is created, the region cannot be changed."
- Bucket Name:** A text input field with a "View naming rules" link. Three constraints are listed: "Cannot be the same as that of the current user's existing buckets.", "Cannot be the same as that of any other user's existing buckets.", and "Cannot be edited after creation."
- Data Redundancy Policy:** Two buttons: "Multi-AZ storage" (selected) and "Single-AZ storage". A warning message states: "This setting can't be changed after the bucket is created. Multi-AZ storage is more expensive, but offers a higher availability." A note below says: "Data is stored in multiple AZs in the same region, improving availability."
- Default Storage Class:** Three buttons: "Standard" (selected), "Warm", and "Cold". Descriptions: "Standard: For frequently accessed data", "Warm: Less expensive, for infrequently accessed data", "Cold: For data accessed once a year". A note below says: "If you do not specify a storage class during object upload, any objects you upload inherit this default storage class."
- Bucket Policies:** Three buttons: "Private" (selected), "Public Read", and "Public Read/Write". A note below says: "Only the bucket owner has full control over the bucket."
- Server-Side Encryption:** A note: "If server-side encryption is enabled, new objects uploaded to this bucket are automatically encrypted. After a bucket is created, you can also change the encryption configuration on the overview page." Two buttons: "SSE-KMS" (selected) and "Disable". A warning message states: "Encryption is recommended to keep data secure."
- Enterprise Project:** A dropdown menu with "--Select--" and a "Create Enterprise Project" link with a help icon.
- Tags:** A note: "It is recommended that you use TMS's predefined tag function to add the same tag to different cloud resources." A link "View predefined tags" and a "Create" icon. Two input fields: "Tag key" and "Tag value".

A green "Create Now" button is located at the bottom right of the form.

- Step 2** Configure bucket parameters.

**Table 2-4** Bucket parameters

Parameter	Description
Region	Geographic area where a bucket resides. For low latency and faster access, select the region nearest to you. Once the bucket is created, its region cannot be changed.

Parameter	Description
Bucket Name	<p>Name of the bucket. A bucket name must be unique across all accounts and regions. Once a bucket is created, its name cannot be changed.</p> <p>According to the globally applied DNS naming rules, an OBS bucket name:</p> <ul style="list-style-type: none"> <li>• Must be unique across all accounts and regions. The name of a deleted bucket can be reused for another bucket or a parallel file system at least 30 minutes later after the deletion.</li> <li>• Must be 3 to 63 characters long. Only lowercase letters, digits, hyphens (-), and periods (.) are allowed.</li> <li>• Cannot start or end with a period (.) or hyphen (-), and cannot contain two consecutive periods (..) or contain a period (.) and a hyphen (-) adjacent to each other.</li> <li>• Cannot be formatted as an IP address.</li> </ul> <p><b>NOTE</b> When you access OBS through HTTPS using virtual hosted-style URLs, if the bucket name contains a period (.), the certificate verification will fail. To work around this issue, you are advised not to use periods (.) in bucket names.</p>
Data Redundancy Policy	<ul style="list-style-type: none"> <li>• <b>Multi-AZ storage:</b> Data is stored in multiple AZs to achieve higher reliability.</li> <li>• <b>Single-AZ storage:</b> Data is stored in a single AZ, with lower costs.</li> </ul> <p>Once a bucket is created, the data redundancy policy cannot be changed, so choose the policy that can meet your needs.</p> <ul style="list-style-type: none"> <li>• Multi-AZ storage is not available for buckets in the Cold storage class.</li> </ul>
Default Storage Class	<p>Storage classes of a bucket. Different storage classes meet different requirements for storage performance and costs.</p> <ul style="list-style-type: none"> <li>• The Standard storage class is for storing a large number of hot files or small files that are frequently accessed (multiple times per month on average) and require quick retrieval.</li> <li>• The Warm storage class is for storing data that is less frequently accessed (less than 12 times per year on average) and requires quick retrieval.</li> <li>• The Cold storage class is for archiving data that is rarely accessed (once a year on average) and has no requirements for quick retrieval.</li> </ul> <p>For details, see <a href="#">Storage Classes Overview</a>.</p>

Parameter	Description
Bucket Policy	Controls read and write permissions for buckets. <ul style="list-style-type: none"><li>● <b>Private:</b> No access beyond the bucket ACL settings is granted.</li><li>● <b>Public Read:</b> Anyone can read objects in the bucket.</li><li>● <b>Public Read and Write:</b> Anyone can read, write, or delete objects in the bucket.</li></ul>
Server-Side Encryption	Select <b>SSE-KMS</b> . For the encryption key type, you can choose <b>Default</b> or <b>Custom</b> . If <b>Default</b> is used, the default key of the current region will be used to encrypt your objects. If there is no such a default key, OBS creates one the first time you upload an object. If <b>Custom</b> is used, you can choose a custom key you created on the KMS console and choose the project where the key belongs to encrypt your objects.  When server-side encryption is enabled for a bucket, you can configure an object to inherit the bucket's KMS encryption settings when you upload the object to the bucket.
Enterprise Project	You can add a bucket to an enterprise project for unified management.  Create an enterprise project on the enterprise project page. The default enterprise project is named <b>default</b> .  On the <b>Enterprise Project Management</b> page, create an enterprise project, create a user group and add users to this group, and then add the user group to the enterprise project. By doing so, users in this user group obtain the operation permissions for the buckets and objects in the enterprise project.  <b>NOTE</b> Only an enterprise account can configure enterprise projects. OBS Viewer and OBS Operator are the fine-grained authorizations of the enterprise project user group in OBS.
Tags	Optional. Tags are used to identify and classify buckets in OBS. Each tag is represented by a key-value pair.  For more information, see <a href="#">Tag Overview</a> .

**Step 3** Click **Create Now**.

----End

## Related Operations

After the bucket is created, you can change its storage class by performing the following steps:

**Step 1** In the bucket list on OBS Console, select the target bucket and click **Change Storage Class** on the right.



**Step 2** Select the desired storage class and click **OK**.

 **NOTE**

- Changing the storage class of a bucket does not change the storage class of existing objects in the bucket.
- If you do not specify a storage class for an object when uploading it, it inherits the bucket's storage class by default. After the bucket's storage class is changed, newly uploaded objects will inherit the new storage class of the bucket by default.

----End

## 2.5.2 Viewing Basic Information of a Bucket

On OBS Console, you can view a bucket's details, including basic bucket information, process flows for common scenarios, domain name details, basic configurations, and others. You can also export all buckets of the current account and view their basic information in the exported Excel file.

### Viewing Bucket Details

- Step 1** In the bucket list, click the bucket you want to operate to go to the **Objects** page.
- Step 2** In the navigation pane, choose **Overview**.
- Step 3** On the top of the page, view the bucket information, including the bucket name, storage class, data redundancy policy, region, and creation time.

**Figure 2-10** Bucket information

[Bucket List](#) / [Objects](#)



test-2024



Standard | Multi-AZ storage | ru-moscow | Created Jan 23, 2024 19:50:24 GMT+08:00

**Table 2-5** Bucket information

Item	Description
Bucket name	Name of the bucket
Storage class	Storage class of the bucket, which can be <b>Standard</b> , <b>Warm</b> , or <b>Cold</b> .
Data redundancy	Data redundancy storage policy of a bucket, which can be multi-AZ storage or single-AZ storage. This setting cannot be changed after the bucket is created.
Region	Region where the bucket is located
Created	Creation time of the bucket

- Step 4** In the **Basic Information** area, view the number of objects, storage usage, bucket version, versioning status, enterprise project, and account ID.

**Figure 2-11** Bucket's basic information



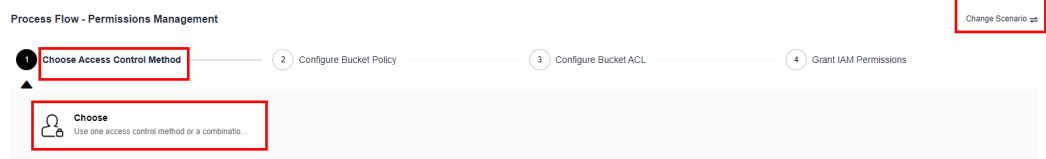
**Table 2-6** Parameters in the Basic Information area

Parameter	Description
Objects	The total number of stored folders and objects of all versions in a bucket
Used Capacity	Total storage space occupied by objects of all versions in the bucket <b>NOTE</b> If usage statistics is available for the bucket, storage usage data is not displayed here.
Bucket Version	Version number of the bucket. <b>3.0</b> indicates the latest bucket version, and -- indicates versions earlier than 3.0.
Versioning	Versioning status
Enterprise Project	Enterprise project where the bucket belongs
Account ID	Unique identifier of the bucket owner, which is the same as the <b>Account ID</b> on the <b>My Credentials</b> page.

**Step 5** In the **Process Flow** area, view the process flows for common scenarios. You can click **Change Scenario** in the upper right corner to choose a desired scenario, as shown in **Figure 2-12**.

In each flow, you can click a node to view relevant details, or click a card to navigate to the operation guide or console page.

**Figure 2-12** Process flows for common scenarios



**Step 6** In the **Domain Information Details** area, view information about the endpoint, access domain name, and static website hosting domain name. You can also perform related operations by clicking buttons in the **Operation** column, as shown in **Figure 2-13**.

**Figure 2-13** Domain name details of the bucket

Domain Name Details

Type	Domain Name	Protocol	Operation
Endpoint <sup>?</sup>	<input type="text"/>	HTTPS/HTTP	--
Access Domain Name <sup>?</sup>	<input type="text"/>	HTTPS/HTTP	Bind User Domain Name
Static website hosting domain name	--	HTTPS/HTTP	Configure

**Step 7** In the **Basic Configurations** area, view the bucket's basic configurations, including lifecycle rules, static website hosting, and CORS rules. You can click a card to make required configurations, as shown in **Figure 2-14**.

**Figure 2-14** Basic configurations of the bucket

Basic Configurations

Lifecycle Rules Not configured >

Static Website Hosting Not configured >

CORS Rules Not configured >

URL Validation Not configured >

Event Notification Not configured >

Tags Not configured >

Logging Not configured >

Default Encryption Not configured >

Versioning Disabled >

----End

## Exporting a Bucket List

**Step 1** Go to the bucket list.

**Step 2** Export all buckets. Specifically, click **Export** in the upper left corner of the bucket list.

**Figure 2-15** Exporting all buckets

**Step 3** Export the selected buckets. Specifically, select the buckets to export and click **Export** in the upper left corner of the bucket list.

**Figure 2-16** Exporting the selected buckets

**Step 4** Obtain the bucket list in Excel, which is automatically downloaded to your local computer.

The file lists all the buckets of the current account and includes the following information: bucket name, storage class, region, data redundancy policy, used capacity, object quantity, bucket version, enterprise project, and bucket creation time.

----End

### 2.5.3 Searching for a Bucket

On OBS Console, you can search for buckets by bucket name, region, storage class, data redundancy policy, and enterprise project.

 **NOTE**

Currently, bucket search by tag is not supported.  
The keywords used for search are case-insensitive.

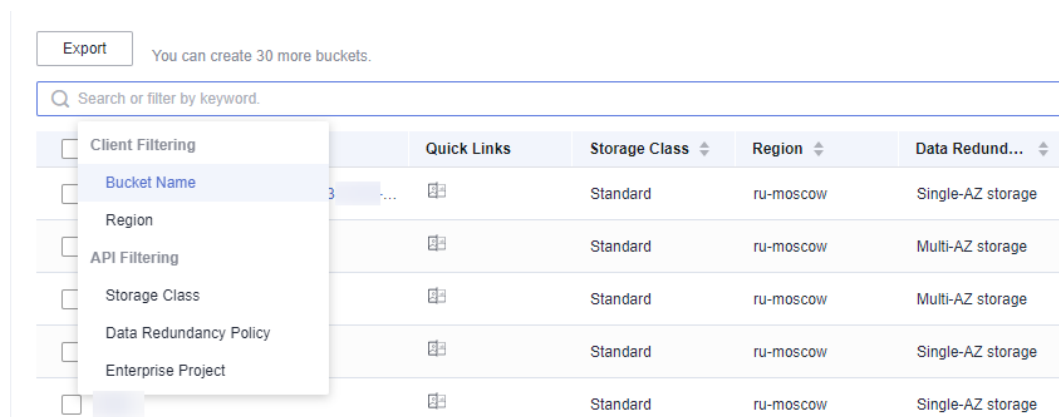
#### Procedure

**Step 1** Click the search box above the bucket list, select **Bucket Name**, **Region**, **Storage Class**, **Data Redundancy Policy**, or **Enterprise Project** from the level-1 drop-down list, and then the option you need from the corresponding level-2 drop-down list. Alternatively, after selecting an option from the level-1 drop-down list, you can enter a keyword in the search box and then select what you want from the level-2 drop-down list.

The found buckets are displayed in the bucket list.


For example, if you want to search for bucket **test**, click the search box, select **Bucket Name** and then **test**. Alternatively, after selecting **Bucket Name**, enter **test** in the search box, and all buckets whose names contain **test** are displayed in the level-2 drop-down list. Then, select **test** and click **OK**.

**Figure 2-17** Searching for buckets




 **NOTE**

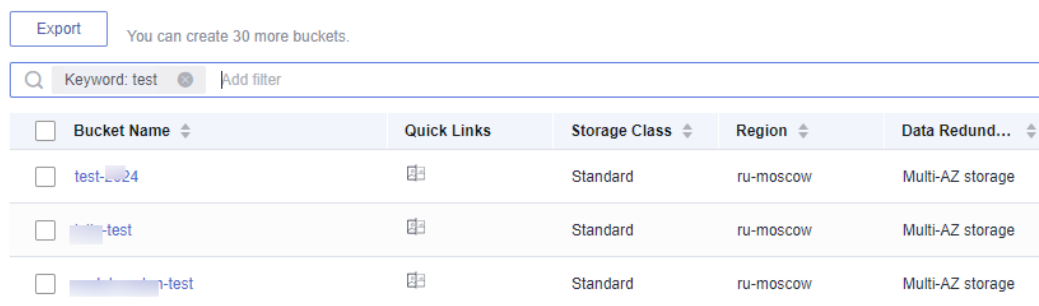
- You can search for buckets based on combinations of different filter criteria.
  - If the filter criteria are of different types, they are in intersection logic.
  - If the filter criteria are of the same type, they are in union logic.
- After a keyword is entered in the search box, all buckets whose name, region, storage class, data redundancy policy, or enterprise project contains the specified keyword are displayed in the drop-down list. Click the option you want. Then, all the buckets meeting the search criteria are displayed in the bucket list.




**Step 2** Enter a keyword in the search box and click  or press **Enter**.

All buckets whose name, region, storage class, data redundancy policy, or enterprise project contains the searched keyword will be displayed in the bucket list.

For example, if you enter **test** in the search box and click  or press **Enter**, all buckets whose name, region, storage class, data redundancy policy, or enterprise project contains keyword **test** are displayed in the bucket list.


**Figure 2-18** Searching for buckets



<input type="checkbox"/> Bucket Name	Quick Links	Storage Class	Region	Data Redund...
<input type="checkbox"/> test-2024		Standard	ru-moscow	Multi-AZ storage
<input type="checkbox"/> test-test		Standard	ru-moscow	Multi-AZ storage
<input type="checkbox"/> test-test		Standard	ru-moscow	Multi-AZ storage

----End

## Related Operations

In the bucket list, click  next to the bucket name, storage class, region, data redundancy policy, used capacity, number of objects, enterprise project, or creation time to sort buckets.

## 2.5.4 Deleting a Bucket

You can delete unwanted buckets on OBS Console to free up the quota of buckets.

### Prerequisites

- All objects in the bucket have been permanently deleted. A bucket must be emptied before it can be deleted.

**NOTICE**

Objects under the **Objects**, **Deleted Objects**, and **Fragments** tabs must be all deleted.

- A bucket can only be deleted by the bucket owner.

## Procedure

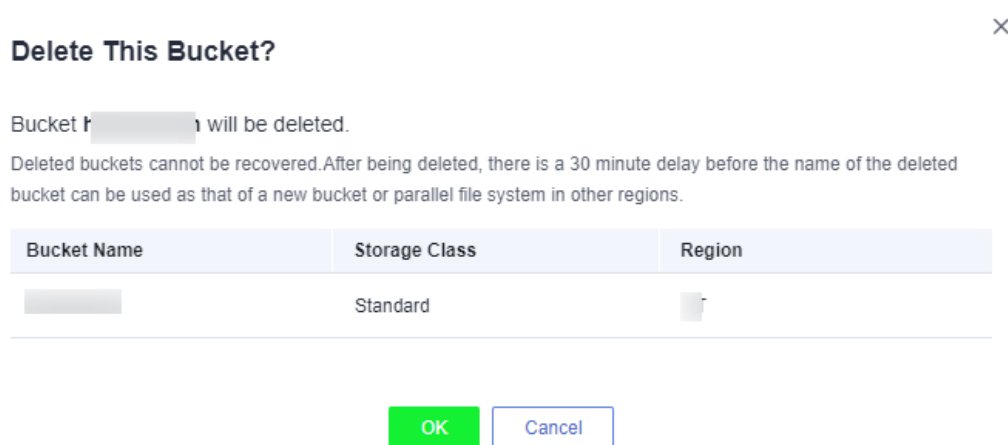
**Step 1** In the bucket list on OBS Console, select the bucket you want to delete, and then click **Delete** on the right.

 **NOTE**

The name of a deleted bucket can be reused for another bucket or parallel file system at least 30 minutes after the deletion.

**Step 2** Click **OK** to confirm the deletion.

**Figure 2-19** Deleting a bucket



----End

## 2.6 Managing Objects

### 2.6.1 Creating a Folder

This section describes how to create a folder on OBS Console. Folders facilitate data management in OBS.

#### Background Information

- Unlike a file system, OBS does not involve the concepts of file and folder. For easy data management, OBS provides a method to simulate folders. In OBS, an object is simulated as a folder by adding a slash (/) to the end of the object name on OBS Console. If you call the API to list objects, paths of objects are returned. In an object path, the content following the last slash (/) is the object name. If a path ends with a slash (/), it indicates that the object is a folder. The hierarchical depth of the object does not affect the performance of accessing the object.
- OBS Console does not support the download of folders. You can use OBS Browser+ to download folders.

## Procedure

**Step 1** In the bucket list, click the bucket you want to operate to go to the **Objects** page.

**Step 2** Click **Create Folder**, or click a folder in the object list to open it and click **Create Folder**.

**Step 3** In the **Folder Name** text box, enter a name for the folder.

- You can create single-level or multi-level folders.
- The name cannot contain the following special characters: \:\*?"<>|
- The name cannot start or end with a period (.) or slash (/).
- The folder's absolute path cannot exceed 1,023 characters.
- Any single slash (/) separates and creates multiple levels of folders at once.
- The name cannot contain two or more consecutive slashes (/).

**Step 4** Click **OK**.

----End

## Follow-up Procedure

You can click **Copy Path** on the right to copy the path of the folder and share it with others. Then they can open the bucket where the folder is stored and enter the path in the search box above the object list to find the folder.

## 2.6.2 Uploading an Object

This section describes how to upload local files to OBS over the Internet. These files can be texts, images, videos, or any other type of files.

### Limitations and Constraints

- OBS Console allows you to upload files in a batch. Up to 100 files can be uploaded at a time, with the total size of no more than 5 GB. If the file size exceeds 5 GB, but no larger than 48.8 TB, use tools (such as OBS Browser+ and obsutil) or the multipart upload of OBS SDKs and APIs for upload.
- If versioning is disabled for your bucket and you upload a new file with the same name as the one you previously uploaded to your bucket, the new file automatically overwrites the previous file and does not retain its ACL information. If you upload a new folder using the same name that was used with a previous folder in the bucket, the two folders will be merged, and files in the new folder will overwrite namesake files in the previous folder.
- After versioning is enabled for your bucket, if the new file you upload has the same name as the one you previously uploaded to the bucket, a new file version will be added in the bucket. For details, see [Versioning Overview](#).

### Prerequisites

- At least one bucket has been created.
- If you want to classify files, you can create folders and upload files to different folders. For details, see [Creating a Folder](#).

## Procedure

**Step 1** In the bucket list, click the bucket you want to operate to go to the **Objects** page.

**Step 2** Go to the folder where you want to upload files and click **Upload Object**. The **Upload Object** dialog box is displayed.

### NOTE

If the files that you want to upload to OBS are stored in Microsoft OneDrive, it is recommended that the names of these files contain a maximum of 32 characters to ensure compatibility.

**Figure 2-20** Uploading objects



**Step 3** Specify the read and write permissions for the object.

**Step 4** Select a storage class. If you do not specify a storage class, the objects you upload inherit the default storage class of the bucket.

### NOTE

An object can have a different storage class from its bucket. You can specify a storage class for an object when uploading it, or you can change the object storage class after the object is uploaded.

**Step 5** In the **Upload Object** area, drag and drop the files or folders you want to upload.

You can also click **add files** in the **Upload Object** area to select files.

**Step 6** **Server-Side Encryption:** Choose **SSE-KMS** or **Disable**. For details, see [Uploading an Object in Server-Side Encryption Mode](#).



**NOTE**

If a bucket has server-side encryption configured, you can select **Inherit from bucket** when uploading an object to the bucket, for the object to inherit the encryption settings from the bucket.

**Step 7** (Optional) To configure metadata, click **Next: (Optional) Configure Advanced Settings**.

Add metadata ContentDisposition, ContentLanguage, WebsiteRedirectLocation, ContentEncoding, or ContentType as needed. For more information, see [Object Metadata](#). Metadata is a set of name-value pairs. The metadata value cannot be left blank. You can add two or more metadata entries by clicking **Add**.

**Figure 2-21** Configuring metadata

The screenshot shows the 'Upload Object' interface. At the top, there is a title 'Upload Object' with a link 'How to Upload a File Larger than 5 GB?' and a close button 'X'. Below the title, there are two steps: '1 Upload Object' (checked) and '2 (Optional) Configure Advanced Settings' (active). The 'Metadata' section contains a description: 'Object metadata is a pair of name and value. Metadata can be used to manage objects. [Learn more](#)'. There are two input fields: 'Metadata name' and 'Metadata value'. Below these fields is a '+ Add' button. At the bottom, there are three buttons: 'Previous: Upload Objects', 'Upload' (green), and 'Cancel'.

**Step 8** Click **Upload**.

----End

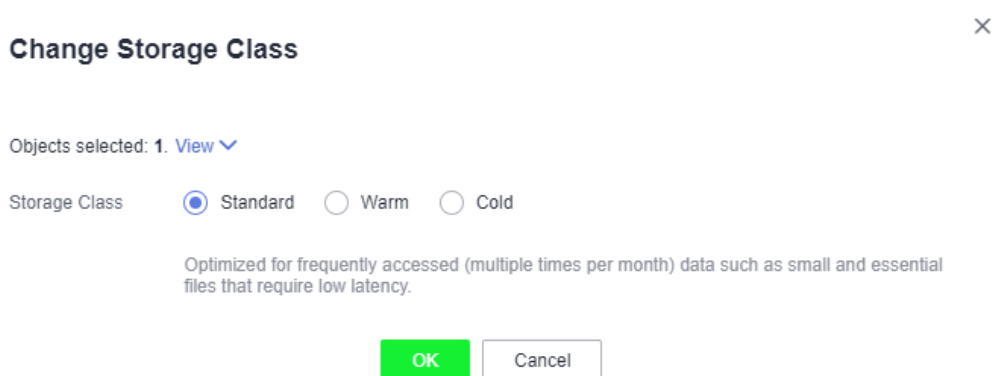
**Related Operations**

When uploading an object, you can specify a storage class for it. After the object is uploaded, you can also change its storage class by doing as follows:

**Step 1** In the bucket list, click the bucket you want to operate to go to the **Objects** page.

**Step 2** Select the target object and choose **More > Change Storage Class** on the right.

**Step 3** Select the desired storage class and click **OK**.

**Figure 2-22** Changing a storage class

----End

**NOTE**

- You can manually change objects between storage classes:
  - From Standard to Warm, or Cold
  - From Warm to Standard, or Cold
  - From Cold to Standard, or Warm. Before changing Cold objects, you must restore them first.  
Changing objects from Warm or Cold to other storage classes incurs restore costs. Select an appropriate change option based on your actual needs.
- After an object is changed to Cold, its restore status changes to **Unrestored**.
- You can also configure a lifecycle rule to change the storage class of an object. For details, see [Configuring a Lifecycle Rule](#).

## Follow-up Procedure

You can click **Copy Path** on the right of an object to copy its path.

You can share the path with others. Then they can open the bucket where the object is stored and enter the path in the search box above the object list to find the object.

## 2.6.3 Downloading an Object

You can download files from OBS Console to the system default path or a custom download path on your local computer.

### Limitations and Constraints

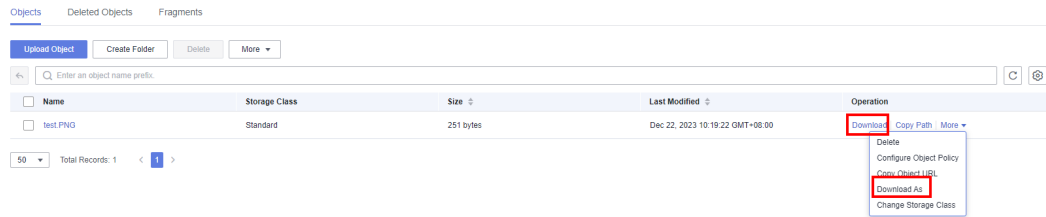
Objects in the Cold storage class can be downloaded only when they are in the **Restored** state.

### Procedure

- Step 1** In the bucket list, click the bucket you want to operate to go to the **Objects** page.
- Step 2** Select the file you want to download. Then, click **Download** or **More > Download As** on the right.

You can also select multiple files and choose **More > Download** above the file list.

**Figure 2-23** Downloading an object



**NOTE**

In the **Download As** dialog box, right-click the object and choose **Copy Link Address** from the shortcut menu to obtain the object's download address.

----End

## 2.6.4 Sharing an Object

### Scenarios

You can share temporary URLs of your objects with others for them to access your objects stored in OBS.

### Background Information

File sharing is temporary. All sharing URLs are only valid for a limited period of time.

A temporary URL consists of the access domain name and the temporary authentication information of a file.

The temporary authentication information contains the **AccessKeyId**, **Expires**, **x-obs-security-token**, and **Signature** parameters. **AccessKeyId**, **x-obs-security-token**, and **Signature** are used for authentication. The **Expires** parameter specifies the validity period of the authentication.

After an object is shared on OBS Console, the system will generate a URL that contains the temporary authentication information, valid for five minutes since its generation by default. Each time you change the validity period of a URL, OBS obtains the authentication information again to generate a new URL for sharing, which takes effect since the time when the validity period is changed.

### Limitations and Constraints

- An object shared from OBS Console can be valid for one minute to 18 hours. If you need a longer validity period, use OBS Browser+ that allows a validity period from one minute to 30 days. Or, you can configure a **bucket policy** or **object policy** to grant other users access to the object permanently.
- Only buckets of version 3.0 support object sharing. You can view the bucket version in the **Basic Information** area on the **Overview** page of a bucket.

- To share a cold object, restore it first.

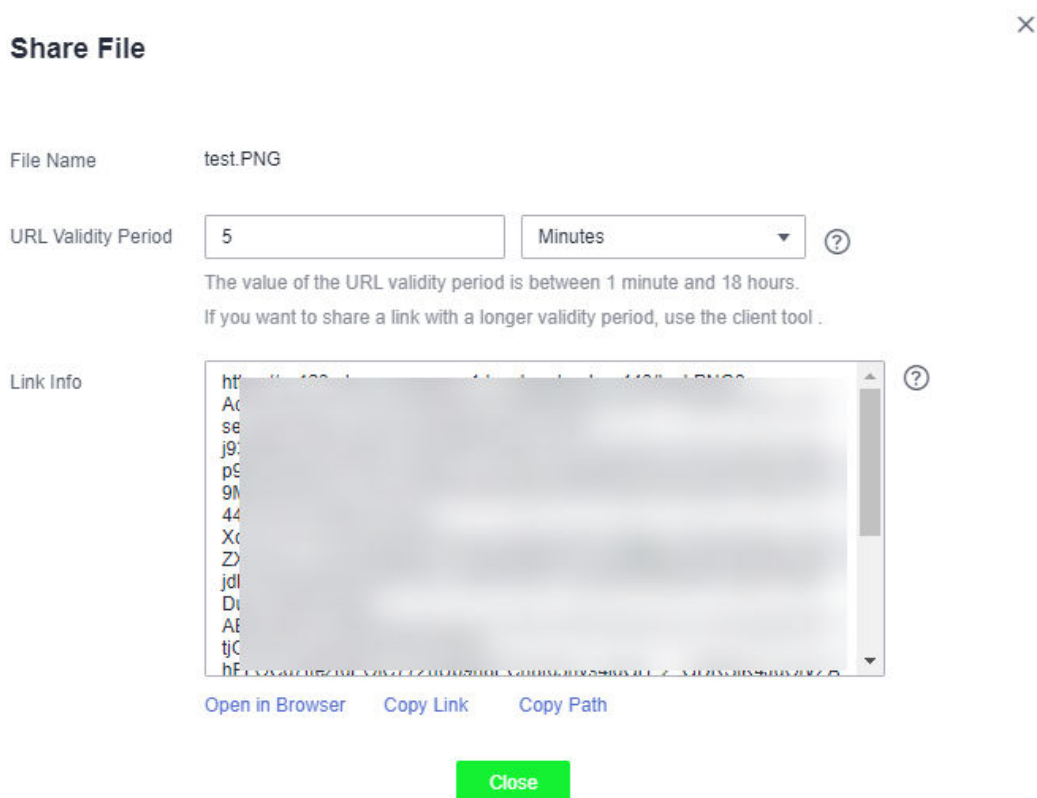
## Procedure

**Step 1** In the bucket list, click the bucket you want to operate to go to the **Objects** page.

**Step 2** Locate the file to be shared and click **Share** in the **Operation** column.

Once the **Share File** dialog box is opened, the URL is effective and valid for five minutes by default. If you change the validity period, the authentication information in the URL changes accordingly, and the URL's new validity period starts upon the change.

**Figure 2-24** Sharing a file



**Step 3** Operate the URL as follows:

- Click **Open URL** to preview the file on a new page or directly download it to your default download path.
- Click **Copy Link** to share the link to others for them to access this file using a browser.
- Click **Copy Path** to share the file path to users who have access to the bucket. The users then can search for the file by pasting the shared path to the search box of the bucket.

### NOTE

Within the URL validity period, anyone who has the URL can access the file.

----End

## 2.6.5 Searching for an Object or Folder

On OBS Console, you can search for files or folders by storage class, last modification time, or object name prefix.

### Procedure

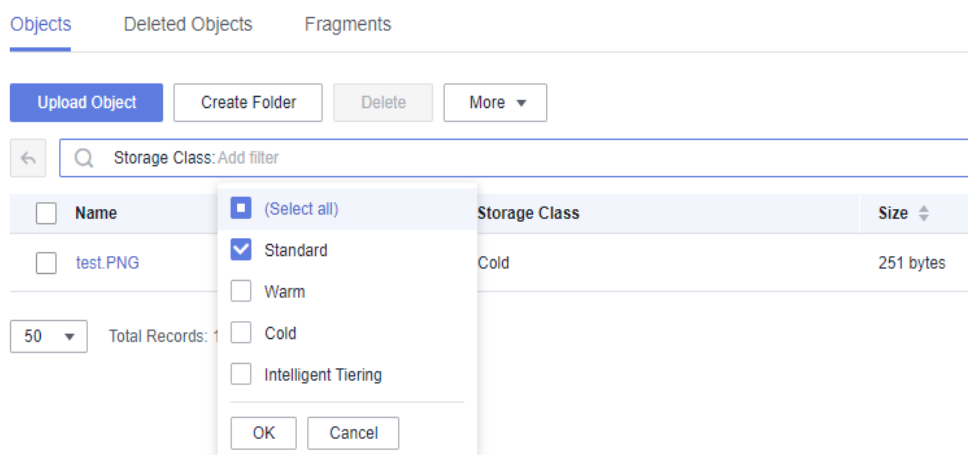
**Step 1** In the bucket list, click the bucket you want to operate to go to the **Objects** page.

**Step 2 Search for objects by storage class:**

1. Click the search box above the object list and choose **Storage Class** from the drop-down list.
2. Select your desired option, or enter a keyword and then select the option displayed.
3. Click **OK**. The searched objects are displayed in the object list.

Suppose you want to search for objects in the Standard storage class. Click the search box and choose **Storage Class** from the drop-down list. Then, select **Standard**, or enter **standard** in the search box and select the option displayed. After that, click **OK**. Objects in the Standard storage class will be displayed in the object list.

**Figure 2-25** Searching for objects by storage class



**Step 3 Search for objects by last modification time:**

1. Click the search box above the object list and choose **Last Modified** from the drop-down list.
2. Select a start date or an end date.

#### NOTE

The time can be accurate to seconds.

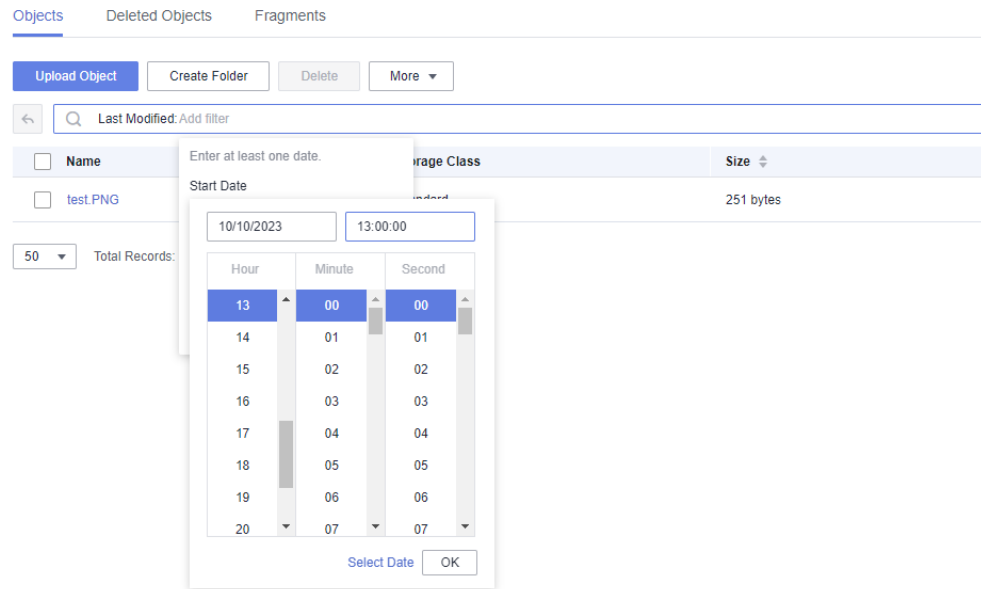
Either start date or end date must be specified.

3. Click **OK**. The objects last modified within the specified time range are displayed in the object list.

Suppose you want to search for objects uploaded from 13:00:00 on October 10, 2023. Click the search box and choose **Last Modified** from the drop-down list.

Then, click the text box for **Start Date**, specify the time (13:00:00 on October 10, 2023) and click **OK**. Objects uploaded from 13:00:00 on October 10, 2023 will be displayed in the object list.

**Figure 2-26** Searching for objects by last modification time



**Step 4 Search for objects by object name prefix:**

1. Click the search box above the object list and choose **Object Name Prefix** from the drop-down list.
2. In the search box, enter the name prefix of the files or folders you want to search for.

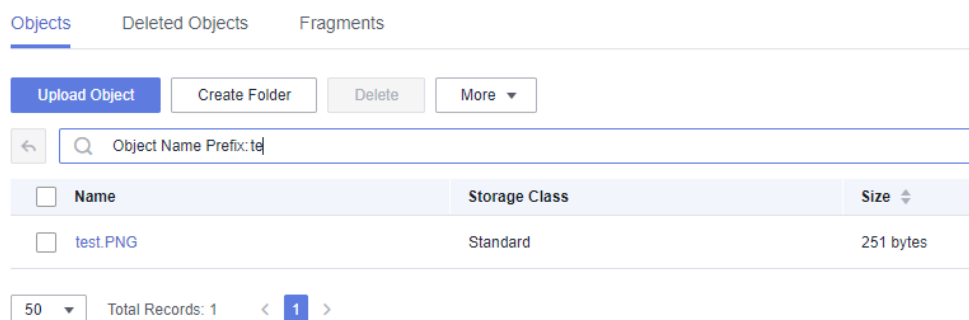
**NOTE**

The object name prefix is case sensitive.

3. Click  or press **Enter**. The searched objects are displayed in the object list.

Suppose you want to search for objects whose prefix is **te**. Click the search box and choose **Object Name Prefix** from the drop-down list box. Then, enter **te**, and press **Enter**. Objects with the **te** prefix will be displayed in the object list.

**Figure 2-27** Searching for objects by object name prefix



 NOTE

To search for objects within a folder, use either of the following methods:

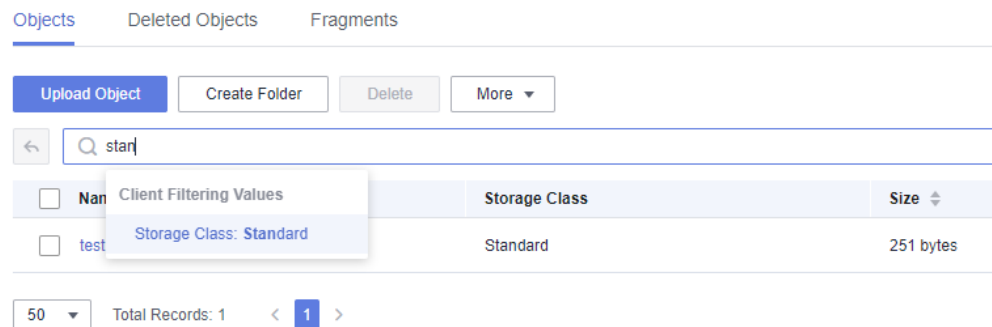
- In the root directory, click the search box above the object list and choose **Object Name Prefix** from the drop-down list. Then, enter *Folder path/Prefix* in the search box. For example, if you enter **abc/123/example**, all files and folders with the **example** prefix in the **abc/123** folder will be displayed.
- Open the folder, and enter the object name prefix in the search box. For example, after you open the **abc/123** folder and enter **example** in the search box, all files and folders with the **example** prefix in the **abc/123** folder will be displayed.

**Step 5** Search for objects by entering a storage class keyword or an object name prefix in the search box above the object list.

1. Enter a storage class keyword. All objects whose storage class contains the specified keyword will be displayed in the object list.

Suppose you enter **Stan** in the search box. The system will then display the **Standard** storage class. After you click this storage class, all objects whose storage class is Standard will be displayed in the object list.

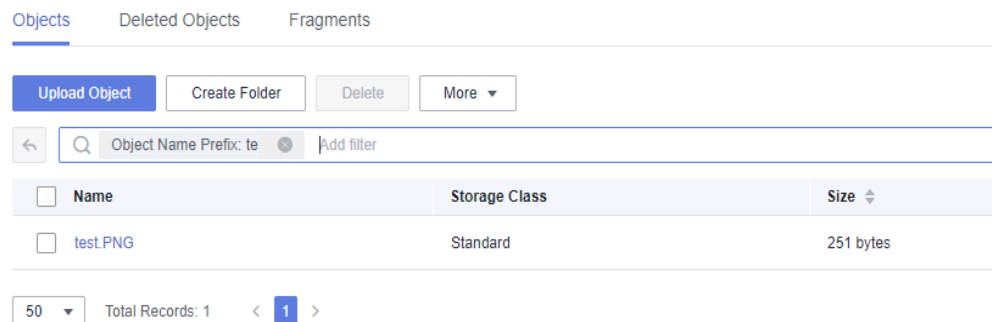
**Figure 2-28** Searching for objects by storage class keyword



2. Enter an object name prefix and click  or press **Enter**. All files and folders with the specified prefix will be displayed in the object list.

Suppose you want to search for objects with the **te** prefix. Enter **te** in the search box and press **Enter**. Then, all objects with the **te** prefix will be displayed in the object list.

**Figure 2-29** Searching for objects by object name prefix




 **NOTE**

You can search for objects based on combinations of different filter criteria.

- If the filter criteria are of different types, they are in intersection logic. For example, if you select storage class **Standard** and object name prefix **te** as two criteria, objects whose storage class is Standard and prefix is **te** will be displayed in the object list.
- If the filter criteria are of the same type, they are in union logic. For example, if you select storage class **Standard** and then **Warm** as two criteria, objects in both Standard and Warm storage classes will be displayed in the object list.

----End

## Related Operations

In the object list, click  next to the size or last modification time to sort objects.

 **NOTE**

Object search by last modification time can only display the first 1,000 records.

If there are more than 5,000 objects in a bucket, the objects are sorted in alphabetical order and can be searched only by object name prefix.

## 2.6.6 Accessing an Object Using Its URL

You can grant anonymous users the read permission for an object so they can access the object using the shared object URL.

### Prerequisites

Anonymous users have the read permission for the object. For details about permission granting, see [Granting Anonymous Users Permission to Access Objects](#).

 **NOTE**

Encrypted objects cannot be shared.

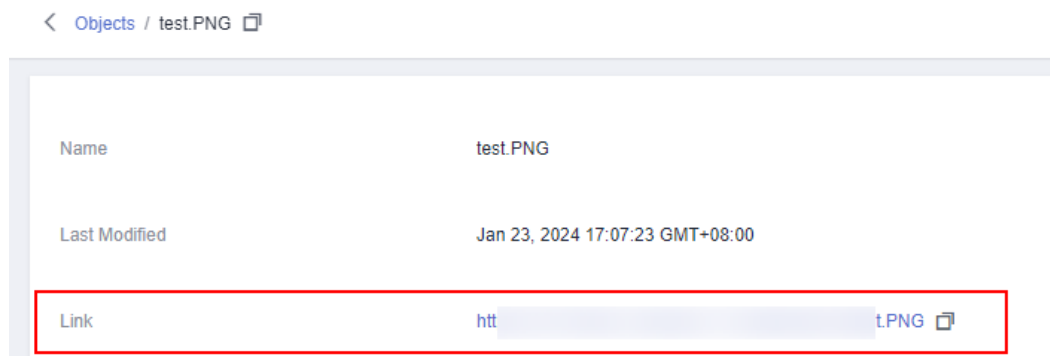
### Procedure

**Step 1** In the bucket list, click the bucket you want to operate to go to the **Objects** page.

**Step 2** Click the object to be shared. The object information is displayed on the top part of the page. You can find the link for accessing the object in the **Link** area.

Anonymous users can access the object by clicking this link. An object link (URL) is in the format of **https://Bucket name.Domain name/Directory level/Object name**. If the object is stored in the root directory of the bucket, its URL does not contain any directory level.



**Figure 2-30** Object link**NOTE**

- To allow anonymous users to access objects in Cold storage using URLs, ensure that these objects are in the **Restored** state.

----End

## 2.6.7 Restoring Objects from the Cold Storage

You must restore a Cold object before you can download it, access it with a URL, or configure its ACL or metadata.

### Limitations and Constraints

- If a Cold object is being restored, its restore task cannot be suspended or deleted.
- An object being restored cannot be restored again.
- After an object is restored, an object copy in the Standard storage class will be generated. This way, there is a Cold object and also its Standard copy in the bucket. The copy will be automatically deleted once the restore expires.

### Procedure

**Step 1** In the bucket list, click the bucket you want to operate to go to the **Objects** page.

**Step 2** Select the file you want to restore, and click **Restore** on the right. The following dialog box shown in **Figure 2-31** is displayed.

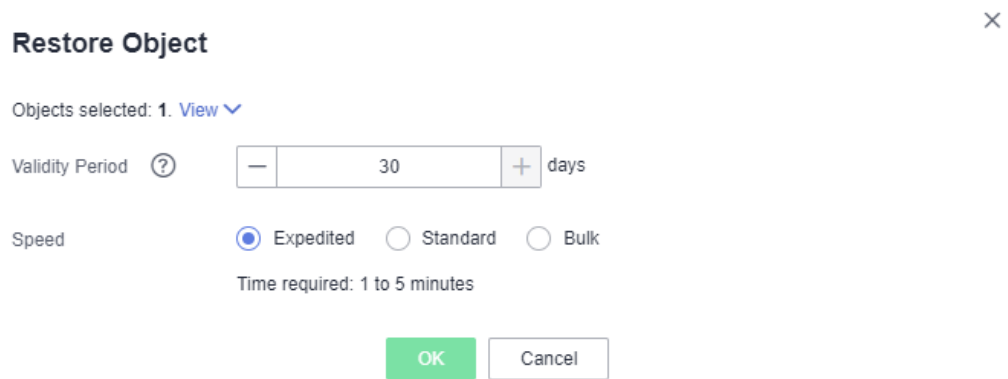
You can select multiple files and click **Restore** above the file list to batch restore the files.

You can select multiple files and choose **More > Restore** above the file list to batch restore them.

**NOTE**

Objects that are being restored cannot be added for batch restore.

**Figure 2-31** Restoring an object



**Step 3** Configure the validity period and speed of the restore. The following table describes the parameters.

**Table 2-7** Parameters for restoring objects

Parameter	Description
Validity Period	How long the object will remain in the <b>Restored</b> state. It starts once the object is restored. The value is an integer ranging from 1 to 30 (days). The default value is <b>30</b> . For example, if you set <b>Validity Period</b> to <b>20</b> when restoring an object, 20 days after the object is successfully restored, its status will change from <b>Restored</b> to <b>Unrestored</b> .
Speed	How fast an object will be restored. <ul style="list-style-type: none"> <li>● <b>Expedited:</b> Cold objects can be restored within 1 to 5 minutes.</li> <li>● <b>Standard:</b> Cold objects can be restored within 3 to 5 hours.</li> <li>● <b>Bulk:</b> Large amounts, even gigabytes, of data can be restored within 5 to 12 hours at a low cost.</li> </ul>

**Step 4** Click **OK**.

**NOTE**

The system checks the file restore status at UTC 00:00 every day. The system starts counting down the expiration time from the time when the latest check is complete.

----End

## Related Operations

Within the validity period of a restored object, you can restore the object again. The validity period is then extended because it will start again when the latest restore is complete.

 NOTE

If a restored object is restored again, its expiration time should be later than the time set for the previous restore. Assume that an object is restored on January 1 and will expire 30 days later (on January 30). If the object is restored again on January 10 and is made to be expired earlier than January 30 (less than 20 days later), this restore action is considered invalid.

## 2.6.8 Deleting an Object or Folder

### Scenarios

On OBS Console, you can manually delete unneeded files or folders to release space and reduce costs.

Alternatively, you can configure lifecycle rules to periodically, automatically delete some or all of the files and folders from a bucket. For details, see [Configuring a Lifecycle Rule](#).

In big data scenarios, parallel file systems usually have deep directory levels and each directory has a large number of files. In such case, deleting directories from parallel file systems may fail due to timeout. To address this problem, you are advised to delete directories in either of the following ways:

1. On the Hadoop client that has OBSA, an OBS client plugin, embedded, run the `hadoop fs -rmr obs://{Name of a parallel file system}/{Directory name}` command.
2. Configure [a lifecycle rule](#) for directories so that they can be deleted in background based on the preset lifecycle rule.

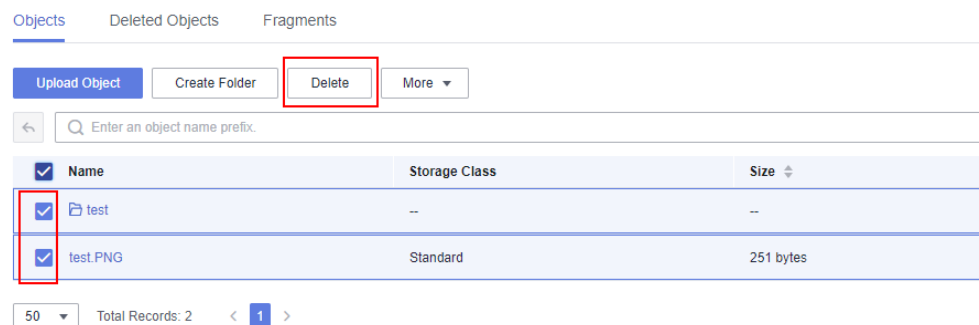
### Background Information

#### Object Deletion with Versioning Enabled

When versioning is enabled for a bucket, OBS works slightly different when deleting different objects.

- Deleting a file or folder: The file or folder is not permanently deleted, but is retained in the **Deleted Objects** list and marked with the **Delete Marker**. In **Deleted Objects**, click the object name. On the **Versions** tab, you can see that the latest object version has the delete marker.

Figure 2-32 Deleting a file or folder



- To permanently delete the file or folder, delete it again from the **Deleted Objects** list. For details, see [Procedure](#).
- To recover the deleted file, undelete it from the **Deleted Objects** list. For details, see [Undeleting an Object](#).
- Deleting an object version: The version will be permanently deleted and cannot be recovered. If the deleted version is the latest one, the next latest version becomes the latest version.

**Figure 2-33** Deleting a version of an object



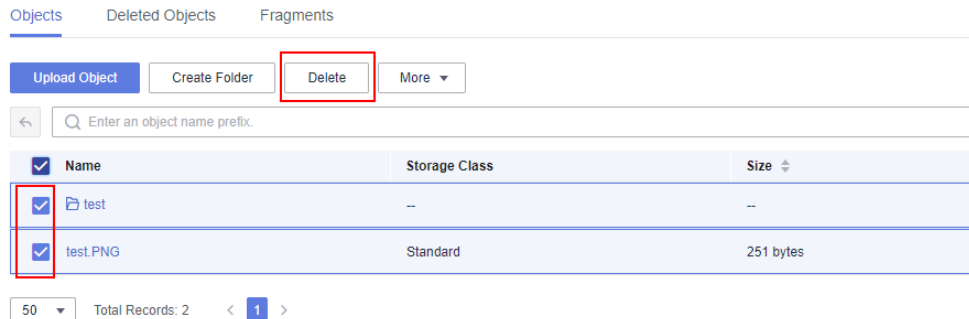
## Procedure

**Step 1** In the bucket list, click the bucket you want to operate to go to the **Objects** page.

**Step 2** Select the file or folder you want to delete and click **Delete** or choose **More > Delete** on the right.

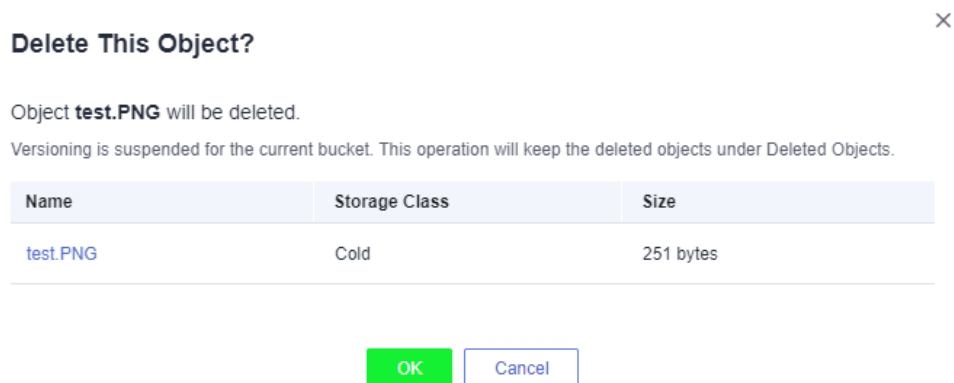
You can select multiple files or folders and click **Delete** above the object list to batch delete them.

**Figure 2-34** Deleting a file or folder



**Step 3** Click **OK** to confirm the deletion.

**Figure 2-35** Deleting objects



**CAUTION**

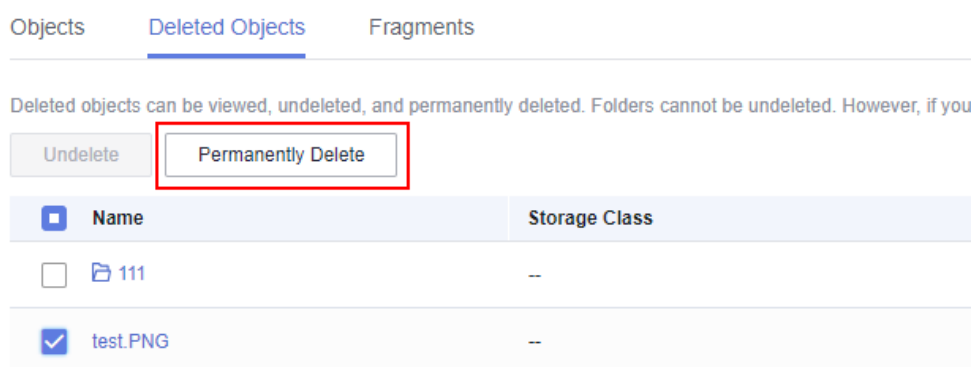
If you delete an object from a bucket with versioning enabled, the object is not permanently deleted. You can view it in the **Deleted Objects** list. The object is still billed for storage. If you need to permanently delete the object, see Step 10.

**Step 4** If versioning is enabled for the bucket, delete the files or folders again from the **Deleted Objects** list to permanently delete them.

1. Click **Deleted Objects**.
2. In the **Operation** column of the file or folder to be deleted, click **Permanently Delete**.

You can also select multiple files or folders and click **Permanently Delete** above the object list to batch delete them.

**Figure 2-36** Deleting a file or folder permanently



----End

## Related Operations

When versioning is enabled, files in the **Deleted Objects** list also have multiple versions. Note the following points when deleting different versions of files:

- Deleting a version with the **Delete Marker** actually recovers this version instead of permanently deleting it. For details, see [Undeleting an Object](#).
- Deleting a version without the **Delete Marker** permanently deletes this version. This version will not be recovered even if the object is recovered later.

## 2.6.9 Undeleting an Object

### Scenarios

If a bucket has [versioning](#) enabled, you can recover a deleted object by undeleting it.

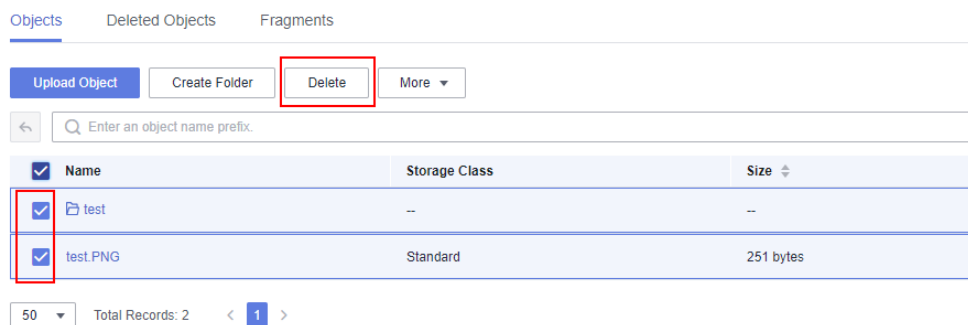
### Background Information

#### Object Deletion with Versioning Enabled

When versioning is enabled for a bucket, OBS works slightly different when deleting different objects.

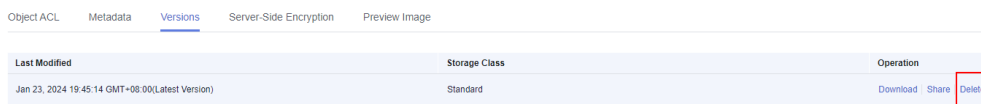
- Deleting a file or folder: The file or folder is not permanently deleted, but is retained in the **Deleted Objects** list and marked with the **Delete Marker**.

**Figure 2-37** Deleting a file or folder



- To permanently delete the file or folder, delete it again from the **Deleted Objects** list. For details, see [Deleting an Object or Folder](#).
- To recover the deleted file, undelete it from the **Deleted Objects** list. For details, see [Procedure](#).
- Deleting an object version: The version will be permanently deleted and cannot be recovered. If the deleted version is the latest one, the next latest version becomes the latest version.

**Figure 2-38** Deleting a version of an object



### Object Recovery with Versioning Enabled

When a bucket has the versioning function enabled, deleting a file from the **Objects** list does not permanently delete it. The deleted file will be retained with the **Delete Marker** in the **Deleted Objects** list. You can recover the deleted file using the **Undelete** operation.

Note the following points when you undelete objects:

1. Only files can be undeleted but not folders.  
After you undelete a deleted file, the file is recovered and will appear in the **Objects** list. Then you can perform basic operations on the file as you normally do on other objects. If the file was stored in a folder before the deletion, it will be recovered to its original path after you undelete it.
2. Deleted files in the **Deleted Objects** also keep multiple versions. When deleting different versions of files, note the following points:
  - If you delete a version with the **Delete Marker**, it actually recovers this version instead of permanently deleting it. For details, see [Related Operations](#).

- If you delete a version without the **Delete Marker**, that version is permanently deleted. This version will not be recovered, even if the object is recovered later.
3. A deleted object must have at least one version without the **Delete Marker** in the **Deleted Objects** list. Otherwise, the object cannot be undeleted.

## Prerequisites

- Versioning has been enabled for the bucket. For details, see [Configuring Versioning](#).
- The file to be recovered is in the **Deleted Objects** list, and has at least one version without the **Delete Marker**.

## Procedure

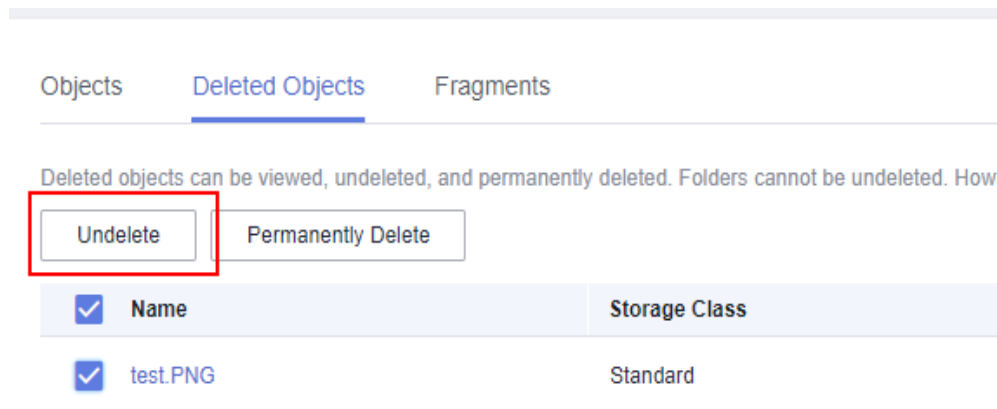
**Step 1** In the bucket list, click the bucket you want to operate to go to the **Objects** page.

**Step 2** Click **Deleted Objects**.

**Step 3** In the row of the deleted object that you want to recover, click **Undelete** on the right.

You can select multiple files and click **Undelete** above the object list to batch recover them.

**Figure 2-39** Undeleting a file



----End

## Related Operations

**Recover a file by deleting its version with the Delete Marker:**

**Step 1** In the bucket list, click the bucket you want to operate to go to the **Objects** page.

**Step 2** Click **Deleted Objects**.

**Step 3** Click the deleted file that you want to recover. The file information is displayed.

**Step 4** On the **Versions** tab, view all versions of the file.

- If you delete a version with the **Delete Marker**, the file will be recovered and retained in the **Objects** list.
- If you delete a version without the **Delete Marker**, that version will be permanently deleted.

----End

## 2.6.10 Managing Fragments

### Background Information

Data can be uploaded to OBS using multipart uploads. There will be fragments generated, if a multipart upload fails because of the following causes (included but not limited to):

- The network is in poor conditions, and the connection to the OBS server is interrupted frequently.
- The upload task is manually suspended.
- The device is faulty.
- The device is powered off suddenly.

On OBS Console, storage used by fragments is charged. Clear fragments when they are not needed. If a file upload task fails, upload the file again.

---

**NOTICE**

Generated fragments take up storage space that is billable.

---

### Procedure

**Step 1** In the bucket list, click the bucket you want to operate to go to the **Objects** page.

**Step 2** Click **Fragments**, select the fragment that you want to delete, and click **Delete** on the right.

You can also select multiple fragments and click **Delete** above the fragment list to batch delete them.

**Step 3** Click **OK** to confirm the deletion.

----End

## 2.7 Server-Side Encryption

### 2.7.1 Server-Side Encryption Overview

After server-side encryption is enabled, objects to be uploaded will be encrypted and stored on the server. When objects are downloaded, they will be decrypted on the server first and then returned in plaintext to you.



Key Management Service (KMS) uses Hardware Secure Modules (HSMs) to ensure key security, enabling users to easily create and manage encryption keys. Keys are not displayed in plaintext outside HSMs, which prevents key disclosure. All operations performed on keys are controlled and logged, and usage of all keys is recorded, meeting regulatory compliance requirements.

The objects to be uploaded can be encrypted from the server side using the encryption service provided by KMS. You need to create a key using KMS or use the default key provided by KMS. Then you can use the key to perform server-side encryption when uploading objects to OBS.

OBS supports both SSE-KMS and server-side encryption with customer-provided keys (SSE-C) by calling APIs. In SSE-C mode, OBS encrypts objects on the server side using the keys and MD5 values provided by customers. Both methods use the AES-256 encryption algorithm.

## 2.7.2 Bucket Default Encryption

OBS allows you to configure default encryption for a bucket. After the default encryption is enabled for the bucket, objects uploaded to this bucket are automatically encrypted using the specified key, making data storage more secure.

You can enable the default encryption (by choosing SSE-KMS) when creating a bucket (see [Creating a Bucket](#)), or enable or disable default encryption after the bucket is created.

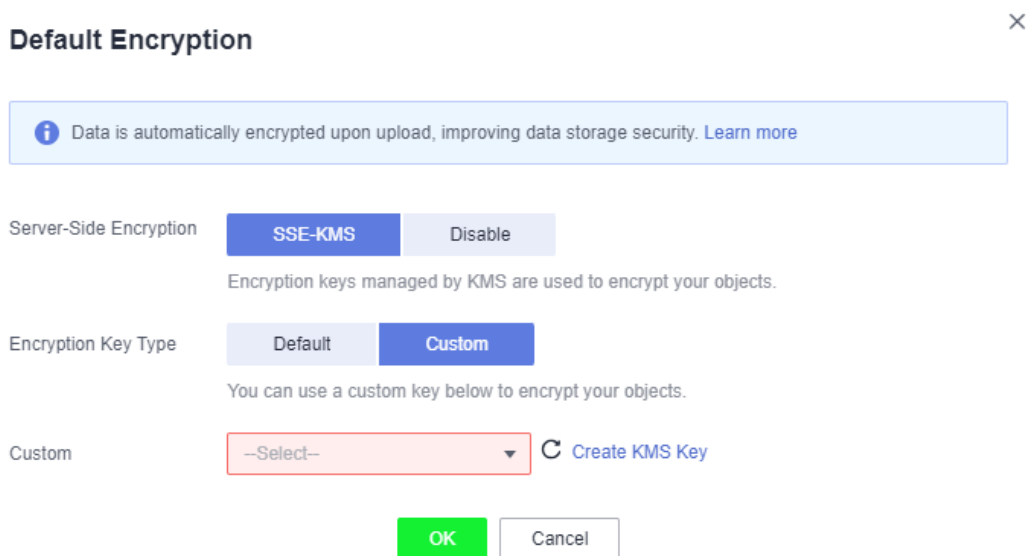
OBS only encrypts the objects uploaded after the default encryption is enabled for the bucket, and does not encrypt those uploaded before. After you disable a bucket's default encryption, the encryption status of existing objects keeps unchanged, and you can separately encrypt objects when uploading them to the bucket.

### Enabling Default Encryption for a Bucket

- Step 1** In the bucket list, click the bucket you want to operate to go to the **Objects** page.
- Step 2** In the navigation pane, choose **Overview**.
- Step 3** In the **Basic Configurations** area, click **Default Encryption**. The **Default Encryption** dialog box is displayed.
- Step 4** Choose **SSE-KMS**.

You can select **Default** to use the default key in the current region to encrypt the objects you upload. If you do not have a default key, OBS automatically creates one the first time you upload an object. You can also choose **Custom** to use a custom key for encryption. If there is no custom key available, click **Create KMS Key** to create one.

**Figure 2-40** Choosing SSE-KMS for a bucket



**Step 5** Click **OK**.

----End

## Disabling Default Encryption for a Bucket

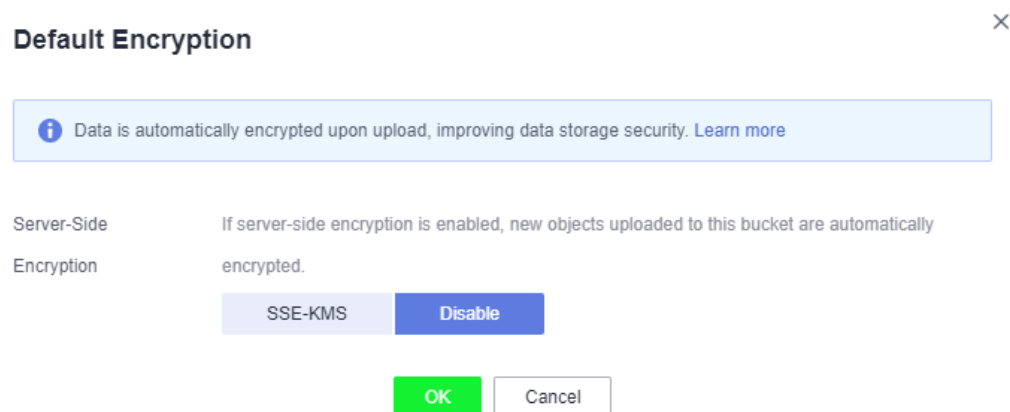
**Step 1** In the bucket list, click the bucket you want to operate to go to the **Objects** page.

**Step 2** In the navigation pane, choose **Overview**.

**Step 3** In the **Basic Configurations** area, click **Default Encryption**. The **Default Encryption** dialog box is displayed.

**Step 4** Select **Disable**.

**Figure 2-41** Disabling encryption for a bucket



**Step 5** Click **OK**.

----End

## 2.7.3 Uploading an Object in Server-Side Encryption Mode

OBS allows you to encrypt objects with server-side encryption so that the objects can be securely stored in OBS.

In a bucket with server-side encryption disabled, objects uploaded to it are not encrypted by default, but you can configure server-side encryption for the objects when uploading them. In a bucket with server-side encryption enabled, objects uploaded to it can inherit the encryption settings of the bucket, and you can also separately configure encryption for the objects.

### Limitations and Constraints

- The object encryption status cannot be changed.
- A key in use cannot be deleted. Otherwise, the object encrypted with this key cannot be downloaded.

### Prerequisites

In the region where OBS is deployed, the **KMS Administrator** permission has been added to the user group. For details about how to add permissions, see the *IAM User Guide*.

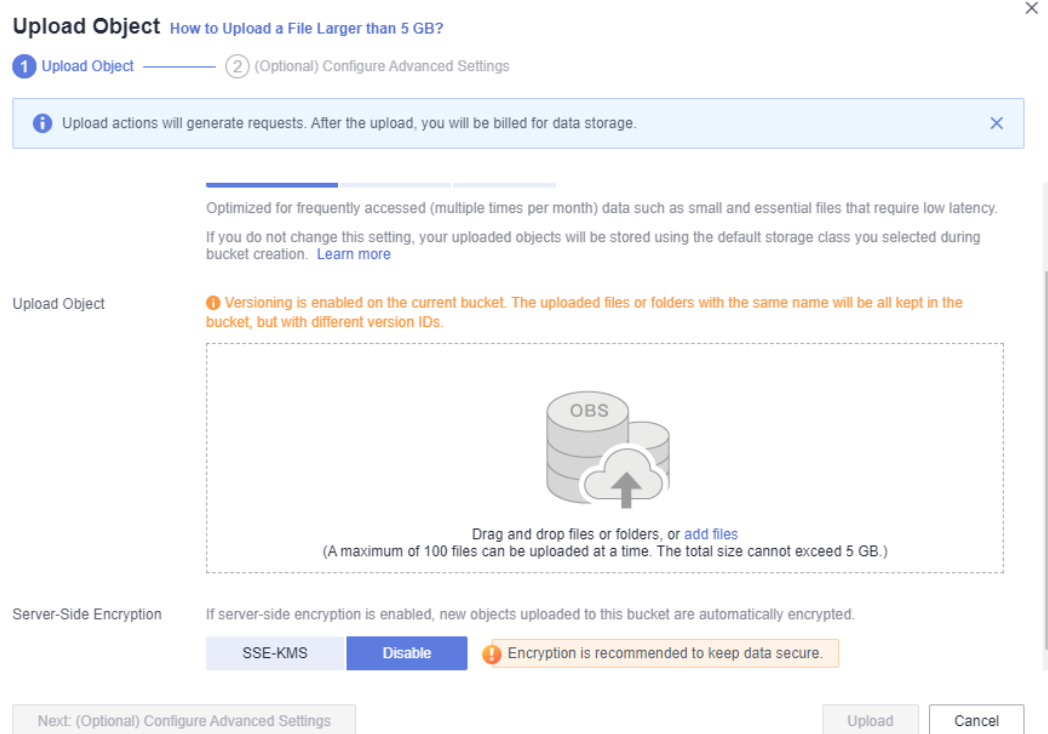
### Procedure

- Step 1** In the bucket list, click the bucket you want to operate to go to the **Objects** page.
- Step 2** Click **Upload Object**. The **Upload Object** dialog box is displayed.
- Step 3** Add the files to be uploaded.
- Step 4** Choose **SSE-KMS**. You can select the default key in the current region to encrypt the objects you upload to the bucket. If you do not have a default key, OBS automatically creates one the first time you upload an object. You can also choose **Custom** to use a custom key for encryption. If there is no custom key available, click **Create KMS Key** to create one.

#### NOTE

When server-side encryption is enabled for a bucket, you can select **Inherit from bucket** when uploading an object, for the object to inherit the encryption settings from the bucket.

Figure 2-42 Encrypting an object to be uploaded



**Step 5** Click **Upload**.

After the object is uploaded, you can view its encryption status on its details page.

----End

## 2.8 Object Metadata

### 2.8.1 Object Metadata Overview

Object metadata is a set of name-value pairs that describe the object and is used for object management.

Currently, only the metadata defined by the system is supported.

The metadata defined by the system is classified into the following types: system-controlled and user-controlled. For example, metadata such as **Last-Modified** is controlled by the system and cannot be modified. You can call the API to modify the metadata such as **ContentLanguage**. The metadata that can be modified is described as follows:

**Table 2-8** OBS metadata

Name	Description
ContentDisposition	<p>Provides a default file name for the object that is being requested. When an object is being downloaded or accessed, the file with the default file name is directly displayed in the browser or a download dialog box is displayed if the file is being accessed.</p> <p>For example, select <b>ContentDisposition</b> as the metadata name and enter <b>attachment;filename="testfile.xls"</b> as the metadata value for an object. If you access the object through a link, a dialog box is directly displayed for downloading objects, and the object name is changed to <b>testfile.xls</b>. For details, see the definition about ContentDisposition in HTTP.</p>
ContentLanguage	<p>Indicates the language or languages intended for the audience. Therefore, a user can differentiate according to the user's preferred language. For details, see the definition about ContentLanguage in HTTP.</p>
WebsiteRedirectLocation	<p>Redirects an object to another object or an external URL. The redirection function is implemented using static website hosting.</p> <p>For example, you can perform the following operations to implement object redirection:</p> <ol style="list-style-type: none"> <li>1. Set metadata of object <b>testobject.html</b> in the root directory of bucket <b>testbucket</b>. Select <b>WebsiteRedirectLocation</b> for <b>Name</b> and enter <b>http://www.example.com</b> for <b>Value</b>.</li> </ol> <p><b>NOTE</b> OBS only supports redirection for objects in the root directory of a bucket. Redirection for objects located in folders of a bucket is not supported.</p> <ol style="list-style-type: none"> <li>2. Configure static website hosting for bucket <b>testbucket</b>, and set the object <b>testobject.html</b> in the bucket as the default home page of the hosted static website.</li> <li>3. If you access object <b>testobject.html</b> through the URL link provided on the <b>Configure Static Website Hosting</b> page, the access request is redirected to <b>http://www.example.com</b>.</li> </ol>

Name	Description
ContentEncoding	Content encoding format when an object is downloaded. The options are as follows: <ul style="list-style-type: none"><li>• Standard: <b>compress</b>, <b>deflate</b>, <b>exi</b>, <b>identity</b>, <b>gzip</b>, and <b>pack200-gzip</b></li><li>• Others: <b>br</b>, <b>bzip2</b>, <b>lzma</b>, <b>peerdist</b>, <b>sdch</b>, <b>xpress</b>, <b>xz</b></li></ul>
CacheControl	Cache behavior of the web page when the specified object is downloaded. <ul style="list-style-type: none"><li>• Cacheability: <b>public</b>, <b>private</b>, <b>no-cache</b>, and <b>only-if-cached</b></li><li>• Expiration time: <b>max-age=&lt;seconds&gt;</b>, <b>s-maxage=&lt;seconds&gt;</b>, <b>max-stale[=&lt;seconds&gt;]</b>, <b>min-fresh=&lt;seconds&gt;</b>, <b>stale-while-revalidate=&lt;seconds&gt;</b>, <b>stale-if-error=&lt;seconds&gt;</b></li><li>• Re-verification and reloading: <b>must-revalidate</b>, <b>proxy-revalidate</b>, <b>immutable</b></li><li>• Others: <b>no-store</b>, <b>no-transform</b></li></ul>
Expires	Cache expiration time (GMT)
ContentType	File type of an object. For details, see <a href="#">About Object Metadata Content-Type</a> .

 NOTE

- When versioning is enabled for a bucket, you can set metadata for objects which are **Latest Version**, but cannot set metadata for objects which are **Historical Version**.
- Metadata cannot be configured for Cold objects.

## 2.8.2 About Object Metadata Content-Type

When an object is uploaded to OBS, the system automatically matches the value of **Content-Type** based on the file name extension of the object. When you access an object through a web browser, the system specifies an application to open the object according to the value of **Content-Type**. You can modify the **Content-Type** of an object based on its file name extension.

**Table 2-9** Common Content-Type values

File Name Extension	Content-Type	File Name Extension	Content-Type
* (binary stream, which does not know the type of the file to be downloaded)	application/octet-stream	.tif	image/tiff
.001	application/x-001	.301	application/x-301
.323	text/h323	.906	application/x-906
.907	drawing/907	.a11	application/x-a11
.acp	audio/x-mei-aac	.ai	application/postscript
.aif	audio/aiff	.aifc	audio/aiff
.aiff	audio/aiff	.anv	application/x-anv
.asa	text/asa	.asf	video/x-ms-asf
.asp	text/asp	.asx	video/x-ms-asf
.au	audio/basic	.avi	video/avi
.awf	application/vnd.adobe.workflow	.biz	text/xml
.bmp	application/x-bmp	.bot	application/x-bot
.c4t	application/x-c4t	.c90	application/x-c90
.cal	application/x-cals	.cat	application/vnd.ms-pki.seccat
.cdf	application/x-netcdf	.cdr	application/x-cdr
.cel	application/x-cel	.cer	application/x-x509-ca-cert
.cg4	application/x-g4	.cgm	application/x-cgm
.cit	application/x-cit	.class	java/*
.cml	text/xml	.cmp	application/x-cmp
.cmx	application/x-cmx	.cot	application/x-cot
.crl	application/pkix-crl	.crt	application/x-x509-ca-cert
.csi	application/x-csi	.css	text/css

File Name Extension	Content-Type	File Name Extension	Content-Type
.cut	application/x-cut	.dbf	application/x-dbf
.dbm	application/x-dbm	.dbx	application/x-dbx
.dcd	text/xml	.dcx	application/x-dcx
.der	application/x-x509-ca-cert	.dgn	application/x-dgn
.dib	application/x-dib	.dll	application/x-msdownload
.doc	application/msword	.dot	application/msword
.drw	application/x-drw	.dtd	text/xml
.dwf	Model/vnd.dwf	.dwf	application/x-dwf
.dwg	application/x-dwg	.dxb	application/x-dxb
.dxf	application/x-dxf	.edn	application/vnd.adobe.edn
.emf	application/x-emf	.eml	message/rfc822
.ent	text/xml	.epi	application/x-epi
.eps	application/x-ps	.eps	application/postscript
.etd	application/x-ebx	.exe	application/x-msdownload
.fax	image/fax	.fdf	application/vnd.fdf
.fif	application/fractals	.fo	text/xml
.frm	application/x-frm	.g4	application/x-g4
.gbr	application/x-gbr	.	application/x-
.gif	image/gif	.gl2	application/x-gl2
.gp4	application/x-gp4	.hgl	application/x-hgl
.hmr	application/x-hmr	.hpg	application/x-hpgl
.hpl	application/x-hpl	.hqx	application/mac-binhex40
.hrf	application/x-hrf	.hta	application/hta
.htc	text/x-component	.htm	text/html



File Name Extension	Content-Type	File Name Extension	Content-Type
.html	text/html	.htt	text/webviewhtml
.htx	text/html	.icb	application/x-icb
.ico	image/x-icon	.ico	application/x-ico
.iff	application/x-iff	.ig4	application/x-g4
.igs	application/x-igs	.iii	application/x-iphone
.img	application/x-img	.ins	application/x-internet-signup
.isp	application/x-internet-signup	.IVF	video/x-ivf
.java	java/*	.jfif	image/jpeg
.jpe	image/jpeg	.jpe	application/x-jpe
.jpeg	image/jpeg	.jpg	image/jpeg
.jpg	application/x-jpg	.js	application/x-javascript
.jsp	text/html	.la1	audio/x-liquid-file
.lar	application/x-laplayer-reg	.latex	application/x-latex
.lavs	audio/x-liquid-secure	.lbm	application/x-lbm
.lmsff	audio/x-la-lms	.ls	application/x-javascript
.ltr	application/x-ltr	.m1v	video/x-mpeg
.m2v	video/x-mpeg	.m3u	audio/mpegurl
.m4e	video/mpeg4	.mac	application/x-mac
.man	application/x-troff-man	.math	text/xml
.mdb	application/msaccess	.mdb	application/x-mdb
.mfp	application/x-shockwave-flash	.mht	message/rfc822
.mhtml	message/rfc822	.mi	application/x-mi
.mid	audio/mid	.midi	audio/mid

File Name Extension	Content-Type	File Name Extension	Content-Type
.mil	application/x-mil	.mml	text/xml
.mnd	audio/x-musicnet-download	.mns	audio/x-musicnet-stream
.mocha	application/x-javascript	.movie	video/x-sgi-movie
.mp1	audio/mp1	.mp2	audio/mp2
.mp2v	video/mpeg	.mp3	audio/mp3
.mp4	video/mp4	.mpa	video/x-mpg
.mpd	application/vnd.ms-project	.mpe	video/x-mpeg
.mpeg	video/mpg	.mpg	video/mpg
.mpga	audio/rn-mpeg	.mpp	application/vnd.ms-project
.mps	video/x-mpeg	.mpt	application/vnd.ms-project
.mpv	video/mpg	.mpv2	video/mpeg
.mpw	application/vnd.ms-project	.mpx	application/vnd.ms-project
.mtx	text/xml	.mxx	application/x-mmxp
.net	image/pnetvue	.nrf	application/x-nrf
.nws	message/rfc822	.odc	text/x-ms-odc
.out	application/x-out	.p10	application/pkcs10
.p12	application/x-pkcs12	.p7b	application/x-pkcs7-certificates
.p7c	application/pkcs7-mime	.p7m	application/pkcs7-mime
.p7r	application/x-pkcs7-certreqresp	.p7s	application/pkcs7-signature
.pc5	application/x-pc5	.pci	application/x-pci
.pcl	application/x-pcl	.pcx	application/x-pcx
.pdf	application/pdf	.pdf	application/pdf

File Name Extension	Content-Type	File Name Extension	Content-Type
.pdx	application/ vnd.adobe.pdx	.pfx	application/x- pkcs12
.pgl	application/x-pgl	.pic	application/x-pic
.pko	application/ vnd.ms-pki.pko	.pl	application/x-perl
.plg	text/html	.pls	audio/scpls
.plt	application/x-plt	.png	image/png
.png	application/x-png	.pot	application/ vnd.ms- powerpoint
.ppa	application/ vnd.ms- powerpoint	.ppm	application/x-ppm
.pps	application/ vnd.ms- powerpoint	.ppt	application/ vnd.ms- powerpoint
.ppt	application/x-ppt	.pr	application/x-pr
.prf	application/pics- rules	.prn	application/x-prn
.prt	application/x-prt	.ps	application/x-ps
.ps	application/ postscript	.ptn	application/x-ptn
.pwz	application/ vnd.ms- powerpoint	.r3t	text/vnd.rn- realtext3d
.ra	audio/vnd.rn- realaudio	.ram	audio/x-pn- realaudio
.ras	application/x-ras	.rat	application/rat- file
.rdf	text/xml	.rec	application/ vnd.rn-recording
.red	application/x-red	.rgb	application/x-rgb
.rjs	application/ vnd.rn- realsystem-rjs	.rjt	application/ vnd.rn- realsystem-rjt
.rlc	application/x-rlc	.rle	application/x-rle

File Name Extension	Content-Type	File Name Extension	Content-Type
.rm	application/ vnd.rn-realmedia	.rmf	application/ vnd.adobe.rmf
.rmi	audio/mid	.rmj	application/ vnd.rn- realsystem-rmj
.rmm	audio/x-pn- realaudio	.rmp	application/ vnd.rn- rn_music_package
.rms	application/ vnd.rn-realmedia- secure	.rmvb	application/ vnd.rn-realmedia- vbr
.rmx	application/ vnd.rn- realsystem-rmx	.rnx	application/ vnd.rn-realplayer
.rp	image/vnd.rn- realpix	.rpm	audio/x-pn- realaudio-plugin
.rsml	application/ vnd.rn-rsml	.rt	text/vnd.rn- realtext
.rtf	application/ msword	.rtf	application/x-rtf
.rv	video/vnd.rn- realvideo	.sam	application/x-sam
.sat	application/x-sat	.sdp	application/sdp
.sdw	application/x-sdw	.sit	application/x- stuffit
.slb	application/x-slb	.sld	application/x-sld
.slk	drawing/x-slk	.smi	application/smil
.smil	application/smil	.smk	application/x-smk
.snd	audio/basic	.sol	text/plain
.sor	text/plain	.spc	application/x- pkcs7-certificates
.spl	application/ futuresplash	.spp	text/xml
.ssm	application/ streamingmedia	.sst	application/ vnd.ms- pki.certstore

File Name Extension	Content-Type	File Name Extension	Content-Type
.stl	application/ vnd.ms-pki.stl	.stm	text/html
.sty	application/x-sty	.svg	text/xml
.swf	application/x- shockwave-flash	.tdf	application/x-tdf
.tg4	application/x-tg4	.tga	application/x-tga
.tif	image/tiff	.tif	application/x-tif
.tiff	image/tiff	.tld	text/xml
.top	drawing/x-top	.torrent	application/x- bittorrent
.tsd	text/xml	.txt	text/plain
.uin	application/x-icq	.uls	text/iuls
.vcf	text/x-vcard	.vda	application/x-vda
.vdx	application/ vnd.visio	.vml	text/xml
.vpg	application/x- vpeg005	.vsd	application/ vnd.visio
.vsd	application/x-vsdx	.vss	application/ vnd.visio
.vst	application/ vnd.visio	.vst	application/x-vst
.vsw	application/ vnd.visio	.vsx	application/ vnd.visio
.vtx	application/ vnd.visio	.vxml	text/xml
.wav	audio/wav	.wax	audio/x-ms-wax
.wb1	application/x-wb1	.wb2	application/x-wb2
.wb3	application/x-wb3	.wbmp	image/ vnd.wap.wbmp
.wiz	application/ msword	.wk3	application/x-wk3
.wk4	application/x-wk4	.wkq	application/x-wkq
.wks	application/x-wks	.wm	video/x-ms-wm

File Name Extension	Content-Type	File Name Extension	Content-Type
.wma	audio/x-ms-wma	.wmd	application/x-ms-wmd
.wmf	application/x-wmf	.wml	text/vnd.wap.wml
.wmv	video/x-ms-wmv	.wmx	video/x-ms-wmx
.wmz	application/x-ms-wmz	.wp6	application/x-wp6
.wpd	application/x-wpd	.wpg	application/x-wpg
.wpl	application/vnd.ms-wpl	.wq1	application/x-wq1
.wr1	application/x-wr1	.wri	application/x-wri
.wrk	application/x-wrk	.ws	application/x-ws
.ws2	application/x-ws	.wsc	text/scriptlet
.wsdl	text/xml	.wvx	video/x-ms-wvx
.xdp	application/vnd.adobe.xdp	.xdr	text/xml
.xfd	application/vnd.adobe.xfd	.xdf	application/vnd.adobe.xdf
.xhtml	text/html	.xls	application/vnd.ms-excel
.xls	application/x-xls	.xlw	application/x-xlw
.xml	text/xml	.xpl	audio/scpls
.xq	text/xml	.xql	text/xml
.xquery	text/xml	.xsd	text/xml
.xsl	text/xml	.xslt	text/xml
.xwd	application/x-xwd	.x_b	application/x-x_b
.sis	application/vnd.symbian.install	.sisx	application/vnd.symbian.install
.x_t	application/x-x_t	.ipa	application/vnd.iphone
.apk	application/vnd.android.package-archive	.xap	application/x-silverlight-app

## 2.8.3 Configuring Object Metadata

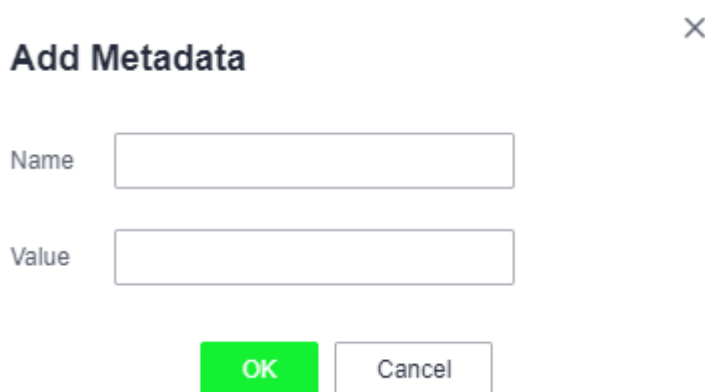
### Procedure

**Step 1** In the bucket list, click the bucket you want to operate to go to the **Objects** page.

**Step 2** Click the object to be operated, and then click the **Metadata** tab.

**Step 3** Click **Add** and specify the metadata information, as shown in [Figure 2-43](#).

**Figure 2-43** Adding metadata



The screenshot shows a dialog box titled "Add Metadata" with a close button (X) in the top right corner. Inside the dialog, there are two input fields: "Name" and "Value". Below the input fields, there are two buttons: "OK" (green) and "Cancel" (white with a grey border).

**Step 4** Click **OK**.

----End

## 2.9 Permissions Control

### 2.9.1 Overview

OBS supports the following permission control mechanisms:

- **IAM policies:** IAM policies define the actions that can be performed on your cloud resources. In other words, IAM policies specify what actions are allowed or denied.
- **Bucket policies and object policies:**  
A bucket policy applies to the configured bucket and objects in the bucket. A bucket owner can use a bucket policy to grant permissions of buckets and objects in the buckets to IAM users or other accounts.  
An object policy applies to specified objects in a bucket.
- **Access control lists (ACLs):** Control the read and write permissions for accounts. You can set ACLs for buckets and objects.

#### NOTE

To test permissions on OBS Console, you need to [create a custom policy](#) to add the IAM user to the user group that has the **obs:bucket:ListAllMyBuckets** permission for all OBS resources. In this way, the IAM user can view the authorized bucket on OBS Console.

## 2.9.2 Permission Control Mechanisms

### 2.9.2.1 IAM Policies

You can create IAM users under a registered cloud service account, and then use IAM policies to control users' access permissions to cloud resources.

IAM policies define the actions that can be performed on your cloud resources. In other words, IAM policies specify what actions are allowed or denied.

IAM policies with OBS permissions take effect on all OBS buckets and objects. To grant an IAM user the permission to operate OBS resources, you need to assign one or more OBS permission sets to the user group to which the user belongs.

For details about OBS permissions controlled by IAM policies, see [Permissions Management](#).

### IAM policies Application Scenarios

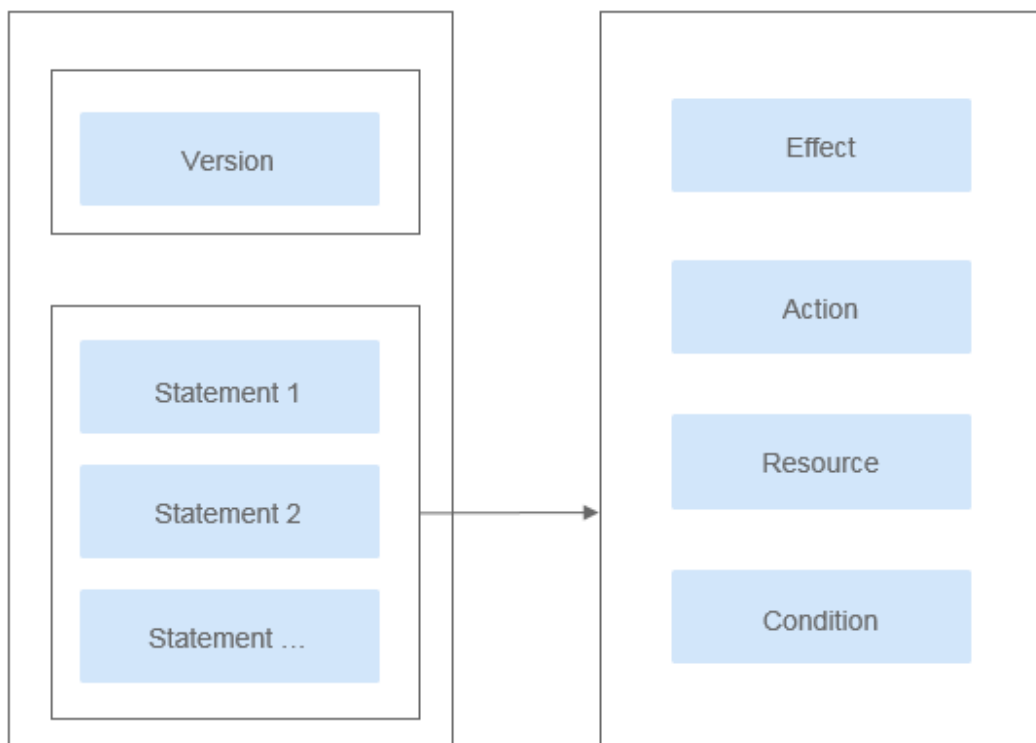
IAM policies are used to authorize IAM users under an account.

- Controlling permissions to cloud resources as a whole under an account
- Controlling permissions to all OBS buckets and objects under an account

### Policy Structure and Syntax

A policy consists of a version and statements. Each policy can have multiple statements.

Figure 2-44 Policy structure





Policy syntax example:

```
{
  "Version": "1.1",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "obs:bucket:HeadBucket",
        "obs:bucket:ListBucket",
        "obs:bucket:GetBucketLocation"
      ],
      "Resource": [
        "obs:*:*:bucket:*"
      ],
      "Condition": {
        "StringEndsWithIfExsits": {
          "g:UserName": ["specialCharacter"]
        },
        "Bool": {
          "g:MFAPresent": ["true"]
        }
      }
    }
  ]
}
```

**Table 2-10** Policy syntax parameters

Parameter	Description
Version	<p>The version number of a policy.</p> <ul style="list-style-type: none"> <li><b>1.0:</b> RBAC policies. An RBAC policy consists of permissions for an entire service. Users in a group with such a policy assigned are granted all of the permissions required for that service.</li> <li><b>1.1:</b> Fine-grained policies. A fine-grained policy consists of API-based permissions for operations on specific resource types. Fine-grained policies, as the name suggests, allow for more fine-grained control on specific operations and resources than RBAC policies. For example: You can restrict an IAM user to access only the objects in a specific directory of an OBS bucket.</li> </ul>

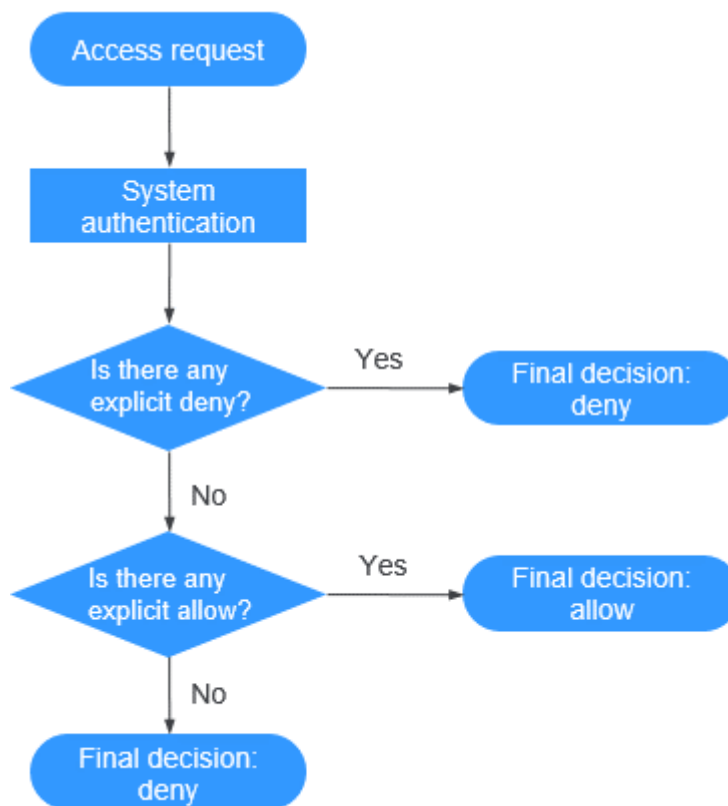
Parameter	Description
Statement	<p>Permissions defined by a policy, including <b>Effect</b>, <b>Action</b>, <b>Resource</b>, and <b>Condition</b>. <b>Condition</b> is optional.</p> <ul style="list-style-type: none"> <li> <b>Effect</b>                      The valid values for <b>Effect</b> are <b>Allow</b> and <b>Deny</b>. System policies contain only <b>Allow</b> statements. For custom policies containing both <b>Allow</b> and <b>Deny</b> statements, the <b>Deny</b> statements take precedence.                 </li> <li> <b>Action</b>                      Permissions of specific operations on resources in the format of <i>Service name.Resource type.Operation</i>. A policy can contain one or more permissions. The wildcard (*) is allowed to indicate all of the services, resource types, or operations depending on its location in the action. OBS has two resource types: bucket and object.                       For details about actions, see the topics of "Bucket-Related Actions" and "Object-Related Actions" in the "IAM Permissions Policies and Supported Actions" section of the <i>OBS API Reference</i>.                 </li> <li> <b>Resource</b>                      Resources on which the policy takes effect in the format of <i>Service name.Region.Domain ID.Resource type.Resource path</i>. The wildcard (*) is allowed to indicate all of the services, regions, resource types, or resource paths depending on its location in the action. In the JSON view, if <b>Resource</b> is not specified, the policy takes effect for all resources.                       The value of <b>Resource</b> supports uppercase (A to Z), lowercase (a to z) letters, digits (0 to 9), and the following characters: -_*.\\. If the value contains invalid characters, use the wildcard character (*).                       OBS is a global service. Therefore, set <b>Region</b> to *. <b>Domain ID</b> indicates the ID of the resource owner. Set it to * to indicate the ID of the account to which the resources belong.                       Examples:                     <ul style="list-style-type: none"> <li>- <b>obs:*:bucket:*</b>: all OBS buckets</li> <li>- <b>obs:*:object:my-bucket/my-object/*</b>: all objects in the <b>my-object</b> directory of the <b>my-bucket</b> bucket</li> </ul> </li> <li> <b>Condition</b>                      Conditions for the policy to take effect (Optional). Format: <i>Condition operator:{Condition key:[Value 1, Value 2]}</i>                       The condition includes the global service condition name and cloud service condition name. The condition names supported by OBS are the same as those in the bucket policy. When configuring in IAM, add <b>obs:</b>. For details, see <a href="#">Conditions</a>.                 </li> </ul>

Parameter	Description
	<p>The value of <b>Condition</b> can contain only uppercase (A to Z), lowercase (a to z) letters, digits (0 to 9), and the following characters: -,./_@#%&amp;. If the value contains unsupported characters, consider using the condition operator for fuzzy match, such as StringLike and StringStartWith.</p> <p>Examples:</p> <ul style="list-style-type: none"> <li>- <b>StringEndWithIfExists</b>:{"g:UserName":["specialCharacter"]}: The statement is valid for users whose names end with <b>specialCharacter</b>.</li> <li>- <b>StringLike</b>:{"obs:prefix":["private/"]}: When listing objects in a bucket, you need to set prefix to <b>private/</b> or include <b>private/</b>.</li> </ul>

### Authentication of IAM policies

The authentication of IAM policies starts from the Deny statements. The following figure shows the authentication logic for resource access.

Figure 2-45 Authentication logic



 **NOTE**

The actions in each policy are in the OR relationship.

1. A user accesses the system and makes an operation request.
2. The system evaluates all the permission policies assigned to the user.
3. In these policies, the system looks for explicit deny permissions. If the system finds an explicit deny that applies, it returns a decision of Deny, and the authentication ends.
4. If no explicit deny is found, the system looks for allow permissions that would apply to the request. If the system finds an explicit allow permission that applies, it returns a decision of Allow, and the authentication ends.
5. If no explicit allow permission is found, IAM returns a decision of Deny, and the authentication ends.

## 2.9.2.2 Bucket Policies and Object Policies

### Bucket Owner and Object Owner

The owner of a bucket is the account that created the bucket. If the bucket is created by an IAM user under the account, the bucket owner is the account instead of the IAM user.

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.

### Bucket Policies

Bucket policies apply to buckets and the objects in them. By leveraging bucket policies, the owner of a bucket can grant IAM users or other accounts the permissions to operate the bucket and objects in the bucket.

#### Application Scenarios

- If no IAM policies are used for access control and you want to grant other accounts the permissions to access your OBS resources, you can use bucket policies.
- You can configure bucket policies to grant IAM users different access permissions on buckets.
- You can also use bucket policies to grant other accounts the permissions to access your buckets.

#### Bucket Policy Templates

OBS Console provides bucket policy templates for eight typical scenarios. You can use these templates to quickly create bucket policies.

Some templates may require a configuration of principals or resources. You can also modify the existing template settings, including principals, resources, actions, and conditions.

**Table 2-11** Bucket policy templates

Principal	Resource	Template	Actions Allowed	Advanced Settings
All accounts	Entire bucket (including the objects in it)	Public Read	<p><b>Allows anonymous users to perform the following actions on a bucket and the objects in it:</b></p> <ul style="list-style-type: none"> <li>GetBucketLocation (to get the bucket location)</li> <li>GetObject (to obtain object content and metadata)</li> <li>RestoreObject (to restore objects from Cold storage)</li> <li>GetObjectVersion (to obtain the content and metadata of a specified object version)</li> </ul>	Excluding the specified actions is not allowed.

Principal	Resource	Template	Actions Allowed	Advanced Settings
		Public Read/Write	<p><b>Allows anonymous users to perform the following actions on a bucket and the objects in it:</b></p> <p>ListBucket (to list objects in the bucket and obtain the bucket metadata)</p> <p>ListBucketVersions (to list object versions in the bucket)</p> <p>HeadBucket (to check whether the bucket exists and obtain the bucket metadata)</p> <p>GetBucketLocation (to get the bucket location)</p> <p>PutObject (to upload objects using PUT and POST, upload parts, initiate multipart uploads, and assemble parts)</p> <p>GetObject (to obtain object content and metadata)</p> <p>ModifyObjectMetaData (to modify object metadata)</p> <p>ListBucketMultipartUploads (to list multipart uploads)</p> <p>ListMultipartUploadParts (to list uploaded parts)</p> <p>AbortMultipartUpload (to abort multipart uploads)</p> <p>RestoreObject (to restore objects from Cold storage)</p> <p>GetObjectVersion (to obtain the content and metadata of a specified object version)</p> <p>PutObjectAcl (to configure the object ACL)</p> <p>GetObjectVersionAcl (to obtain the ACL of a specified object version)</p> <p>GetObjectAcl (to obtain the object ACL)</p>	<p>Excluding the specified actions is not allowed.</p>

Principal	Resource	Template	Actions Allowed	Advanced Settings
Current account/ Other accounts/ Delegated accounts	Entire bucket (including the objects in it)	Bucket Read-Only	<p><b>Allows specified accounts to perform the following actions on a bucket and the objects in it:</b></p> <p>Get* (all GET actions)</p> <p>List* (all LIST actions)</p> <p>HeadBucket (to check whether the bucket exists and obtain the bucket metadata)</p>	Excluding the specified actions is not allowed.
		Bucket Read/Write	<p><b>Allows specified accounts to perform all actions excluding the following ones on a bucket and the objects in it:</b></p> <p>DeleteBucket (to delete the bucket)</p> <p>PutBucketPolicy (to configure a bucket policy)</p> <p>PutBucketAcl (to configure the bucket ACL)</p>	The specified actions are excluded.

Principal	Resource	Template	Actions Allowed	Advanced Settings
All accounts/ Current account/ Other accounts/ Delegated accounts	Current bucket + Specified objects	Directory Read-Only	<p><b>Allows anonymous users or specified accounts to perform the following actions on the current bucket and the specified resources in it:</b></p> <p>GetObject (to obtain object content and metadata)</p> <p>GetObjectVersion (to obtain the content and metadata of a specified object version)</p> <p>GetObjectVersionAcl (to obtain the ACL of a specified object version)</p> <p>GetObjectAcl (to obtain the object ACL)</p> <p>RestoreObject (to restore objects from Cold storage)</p> <p>ListBucket (to list objects in the bucket and obtain the bucket metadata)</p> <p>ListBucketVersions (to list object versions in the bucket)</p> <p>HeadBucket (to check whether the bucket exists and obtain the bucket metadata)</p> <p>GetBucketLocation (to get the bucket location)</p> <p><b>NOTE</b> If you apply the policy to <b>All accounts</b>, <b>ListBucket</b> and <b>ListBucketVersions</b> are not included in the template.</p>	Excluding the specified actions is not allowed.



Principal	Resource	Template	Actions Allowed	Advanced Settings
		Directory Read/Write	<p><b>Allows anonymous users or specified accounts to perform the following actions on the current bucket and the specified resources in it:</b></p> <p>PutObject (to upload objects using PUT and POST, upload parts, initiate multipart uploads, and assemble parts)</p> <p>GetObject (to obtain object content and metadata)</p> <p>GetObjectVersion (to obtain the content and metadata of a specified object version)</p> <p>ModifyObjectMetaData (to modify object metadata)</p> <p>ListBucketMultipartUploads (to list multipart uploads)</p> <p>ListMultipartUploadParts (to list uploaded parts)</p> <p>AbortMultipartUpload (to abort multipart uploads)</p> <p>GetObjectVersionAcl (to obtain the ACL of a specified object version)</p> <p>GetObjectAcl (to obtain the object ACL)</p> <p>PutObjectAcl (to configure the object ACL)</p> <p>RestoreObject (to restore objects from Cold storage)</p> <p>ListBucket (to list objects in the bucket and obtain the bucket metadata)</p> <p>ListBucketVersions (to list object versions in the bucket)</p> <p>HeadBucket (to check whether the bucket exists and obtain the bucket metadata)</p> <p>GetBucketLocation (to get the bucket location)</p>	Excluding the specified actions is not allowed.

Principal	Resource	Template	Actions Allowed	Advanced Settings
All accounts/ Current account/ Other accounts/ Delegated accounts	Specified objects	Object Read-Only	<p><b>Allows anonymous users or specified accounts to perform the following actions on specified resources in the bucket:</b></p> <p>GetObject (to obtain object content and metadata)</p> <p>GetObjectVersion (to obtain the content and metadata of a specified object version)</p> <p>GetObjectVersionAcl (to obtain the ACL of a specified object version)</p> <p>GetObjectAcl (to obtain the object ACL)</p> <p>RestoreObject (to restore objects from Cold storage)</p>	Excluding the specified actions is not allowed.
		Object Read/Write	<p><b>Allows anonymous users or specified accounts to perform the following actions on specified resources in the bucket:</b></p> <p>PutObject (to upload objects using PUT and POST, upload parts, initiate multipart uploads, and assemble parts)</p> <p>GetObject (to obtain object content and metadata)</p> <p>GetObjectVersion (to obtain the content and metadata of a specified object version)</p> <p>ModifyObjectMetaData (to modify object metadata)</p> <p>ListMultipartUploadParts (to list uploaded parts)</p> <p>AbortMultipartUpload (to abort multipart uploads)</p> <p>GetObjectVersionAcl (to obtain the ACL of an object version)</p> <p>GetObjectAcl (to obtain the object ACL)</p> <p>PutObjectAcl (to configure the object ACL)</p> <p>RestoreObject (to restore objects from storage)</p>	Excluding the specified actions is not allowed.

### Custom Bucket Policies

You can also customize bucket policies based on your needs. A custom bucket policy consists of five basic elements: effect, principals, resources, actions, and conditions. For details, see [Bucket Policy Parameters](#).

## Object Policies

Object policies apply to objects in a bucket. A bucket policy is applicable to a set of objects (with the same object name prefix) or to all objects (specified by an asterisk \*) in the bucket. To configure an object policy, select an object, and then configure a policy for it.

### Object Policy Templates

OBS Console provides object policy templates for two typical scenarios. You can use these templates to quickly create object policies.

Some templates may require a configuration of principals. You can also modify the existing template settings, including principals, actions, and conditions. The resource in an object policy is the object that the policy is applied to, which is automatically specified by the system and does not need to be modified.

**Table 2-12** Object policy templates

Principal	Resource	Template	Actions Allowed	Advanced Settings
All accounts/ Current account/ Other accounts/ Delegated accounts	Specified objects	Object Read-Only	<p><b>Allows anonymous users or specified accounts to perform the following actions on specified resources in the bucket:</b></p> <p>GetObject (to obtain object content and metadata)</p> <p>GetObjectVersion (to obtain the content and metadata of a specified object version)</p> <p>GetObjectVersionAcl (to obtain the ACL of a specified object version)</p> <p>GetObjectAcl (to obtain the object ACL)</p> <p>RestoreObject (to restore objects from Cold storage)</p>	Excluding the specified actions is not allowed.

Principal	Resource	Template	Actions Allowed	Advanced Settings
		Object Read/Write	<p><b>Allows anonymous users or specified accounts to perform the following actions on specified resources in the bucket:</b></p> <p>PutObject (to upload objects using PUT and POST, upload parts, initiate multipart uploads, and assemble parts)</p> <p>GetObject (to obtain object content and metadata)</p> <p>GetObjectVersion (to obtain the content and metadata of a specified object version)</p> <p>ModifyObjectMetaData (to modify object metadata)</p> <p>ListMultipartUploadParts (to list uploaded parts)</p> <p>AbortMultipartUpload (to abort multipart uploads)</p> <p>GetObjectVersionAcl (to obtain the ACL of a specified object version)</p> <p>GetObjectAcl (to obtain the object ACL)</p> <p>PutObjectAcl (to configure the object ACL)</p> <p>RestoreObject (to restore objects from Cold storage)</p>	Excluding the specified actions is not allowed.

### Custom Object Policies

You can also customize object policies based on your needs. A custom object policy consists of five basic elements: effect, principals, resources, actions, and conditions, similar to a bucket policy. For details, see [Bucket Policy Parameters](#).

### 2.9.2.3 Bucket ACLs and Object ACLs

Access control lists (ACLs) enable you to manage access to buckets and objects, and define grantees and their granted access permissions. Each bucket and object has its own ACL that defines which accounts or groups are granted access and the type of access. When a request is received against a resource, OBS checks the ACL of the resource to verify whether the requester has necessary access permissions.

When you create a bucket or an object, OBS creates a default ACL that grants the resource owner full control (FULL\_CONTROL) over the bucket or object.

An ACL supports up to 100 grants.

## Who Is a Principal?

A principal can be an account or one of the predefined OBS groups. For details, see [Table 2-13](#).

**Table 2-13** Users supported by OBS

Principal	Description
Specific User	<p>You can grant accounts access permissions to a bucket or an object using ACLs. Once a specific account is granted the access permissions, all IAM users who have OBS resource permissions under this account can have the same access permissions to operate the bucket or object.</p> <p>If you need to grant different access permissions to different IAM users, configure bucket policies. For details, see <a href="#">Granting an IAM User Permissions to Operate a Specific Bucket</a>.</p>
Owner	<p>The owner of a bucket is the account that created the bucket. The bucket owner has all bucket access permissions by default. The read and write permissions for the bucket ACL are permanently available to the bucket owner, and cannot be modified.</p> <p>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. The object owner has the read access to the object, as well as the read and write access to the object ACL, and such access permissions cannot be modified.</p> <p><b>NOTICE</b> Do not modify the bucket owner's read and write access permissions for the bucket.</p>
Anonymous User	<p>If anonymous users are granted access to a bucket or an object, anyone can access the object or bucket without identity authentication.</p>
Log Delivery User	<p>A log delivery user only delivers access logs of buckets and objects to the specified target bucket. OBS does not create or upload any file to a bucket automatically. Therefore, if you want to record bucket access logs, you need to grant the permission to the log delivery user who will deliver the access logs to your specified target bucket. The user only delivers logs within the service scope of OBS.</p> <p><b>NOTE</b> Only the bucket ACL supports authorizing permissions to the log delivery user.</p> <p><b>NOTICE</b> After logging is enabled, the log delivery user group will be automatically granted the permission to read the bucket ACL and write the bucket where logs are saved. If you manually disable such permissions, bucket logging fails.</p>

## What Permissions Can I Grant Using an ACL?

**Table 2-14** lists the permissions you can grant using a bucket ACL.

**Table 2-14** Access permissions controlled by a bucket ACL

Permission	Option	Description
Access to Bucket	READ	Used to obtain the list of objects or object versions in a bucket and to obtain the multipart uploads, metadata, and versioning settings of a bucket.
	WRITE	Used to upload, overwrite, and delete any object in a bucket.
Access to Object	Object READ	Used to obtain the object content and metadata.
Access to ACL	READ_ACL	Used to list the ACLs of a bucket and of objects in the bucket. The bucket owner has this permission permanently by default.
	WRITE_ACL	Used to update the ACL of a bucket. The bucket owner has this permission permanently by default.

**Table 2-15** lists the permissions you can grant using an object ACL.

**Table 2-15** Access permissions controlled by an object ACL

Permission	Option	Description
Access to Object	READ	Used to obtain the content and metadata of an object.
Access to ACL	READ_ACL	Used to obtain the ACL of an object. The object owner has this permission permanently by default.
	WRITE_ACL	Used to update the ACL of an object. The object owner has this permission permanently by default.

### NOTE

Every time you change the bucket or object access permission setting in an ACL, it overwrites the existing setting instead of adding a new access permission to the bucket or object.

You can also set an ACL through a header when invoking the API for creating a bucket or uploading an object. Six types of predefined permissions can be set.

Even with the predefined permissions configured, the bucket or object owner still has the full control over the resource. [Table 2-16](#) lists the predefined permissions.

**Table 2-16** Predefined access permissions in OBS

Predefined Access Permission	Description
private	Indicates that the owner of a bucket or an object has the full control over the resource. Any other users cannot access the bucket or object. This is the default access control policy.
public-read	If this permission is granted on a bucket, anyone can obtain the object list, multipart tasks, metadata, and object versions in the bucket. If it is granted on an object, anyone can obtain the content and metadata of the object.
public-read-write	If this permission is granted on a bucket, anyone can obtain the object list, multipart tasks, metadata, and object versions in the bucket, and can upload or delete objects, initialize multipart upload tasks, upload parts, merge parts, copy parts, and cancel multipart upload tasks. If it is granted on an object, anyone can obtain the content and metadata of the object.
public-read-delivered	If this permission is granted on a bucket, anyone can obtain the object list, multipart tasks, metadata, and object versions, and obtain the object content and metadata in the bucket. It does not apply to objects.
public-read-write-delivered	If this permission is granted on a bucket, anyone can obtain the object list, multipart tasks, metadata, and object versions in the bucket, and can upload or delete objects, initialize multipart upload tasks, upload parts, merge parts, copy parts, and cancel multipart upload tasks. You can also obtain object content and metadata in the bucket. It does not apply to objects.
bucket-owner-full-control	If this permission is granted on a bucket, the bucket can be accessed only by its owner. If it is granted on an object, only the bucket or object owner has the full control over the object.

## Bucket ACL Application Scenarios

ACLs control the read and write permissions for accounts and groups. ACL permission granularity is not as fine as bucket policies and IAM policies. Generally, it is recommended that you use IAM policies and bucket policies for access control.

You can configure bucket ACLs to:

- Grant an account read and write access to a bucket, so that data in the bucket can be shared.

## Object ACL Application Scenarios

ACLs control the read and write permissions for accounts and groups. ACL permission granularity is not as fine as bucket policies and IAM policies. Generally, it is recommended that you use IAM policies and bucket policies for access control.

It is recommended that you use object ACLs in the following scenarios:

- 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.

### 2.9.2.4 Relationship Between a Bucket ACL and a Bucket Policy

#### Mapping Between Bucket ACLs and Bucket Policies

Bucket ACLs are used to control basic read and write access to buckets. Custom settings of bucket policies support more actions that can be performed on buckets. Bucket policies supplement bucket ACLs. In most cases (granting permissions to log delivery user groups excluded), you can use bucket policies to manage access to buckets. [Table 2-17](#) shows the mapping between bucket ACL access permissions and bucket policy actions.

**Table 2-17** Mapping between bucket ACL access permissions and bucket policy actions

ACL Permission	Option	Mapped Action in a Custom Bucket Policy
Access to bucket	Read	<ul style="list-style-type: none"><li>• HeadBucket</li><li>• ListBucket</li><li>• ListBucketVersions</li><li>• ListBucketMultipartUploads</li></ul>



ACL Permission	Option	Mapped Action in a Custom Bucket Policy
	Write	<ul style="list-style-type: none"> <li>PutObject</li> <li>DeleteObject</li> <li>DeleteObjectVersion</li> </ul>
Access to object	Object read	<ul style="list-style-type: none"> <li>GetObject</li> </ul>
Access to ACL	Read	<ul style="list-style-type: none"> <li>GetBucketAcl</li> </ul>
	Write	<ul style="list-style-type: none"> <li>PutBucketAcl</li> </ul>

## Mapping Relationship Between Object ACLs and Bucket Policies

Object ACLs are used to control basic read and write access permissions for objects. The custom settings of bucket policies support more actions that can be performed on objects. [Table 2-18](#) describes the mapping relationship between object ACL access permissions and bucket policy actions.

**Table 2-18** Mapping relationship between object ACLs and bucket policies

Object ACL	Option	Mapped Action in a Custom Bucket Policy
Access to Object	Read	<ul style="list-style-type: none"> <li>GetObject</li> <li>GetObjectVersion</li> </ul>
Access to ACL	Read	<ul style="list-style-type: none"> <li>GetObjectAcl</li> <li>GetObjectVersionAcl</li> </ul>
	Write	<ul style="list-style-type: none"> <li>PutObjectAcl</li> <li>PutObjectVersionAcl</li> </ul>

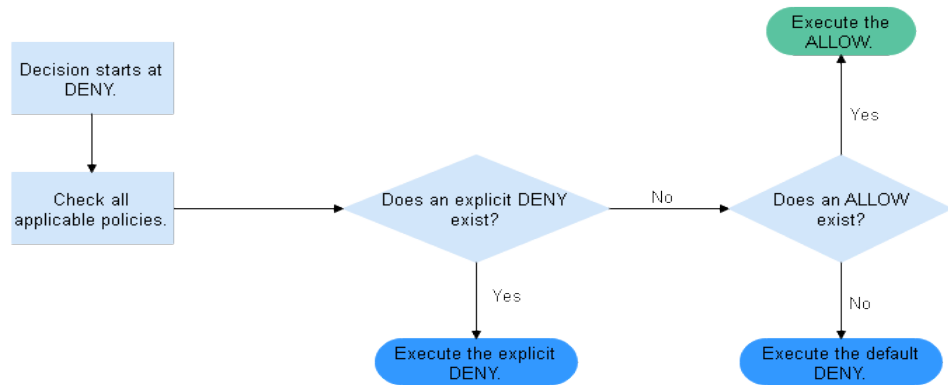
### 2.9.2.5 How Does Authorization Work When Multiple Access Control Mechanisms Co-Exist?

Based on the principle of least privilege, the default access control result is always deny, and an explicit deny statement always take precedence over an allow statement. Suppose that IAM policies 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 user's access to the object 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 2-46** Authorization process



**Figure 2-47** is a matrix of the IAM policies, bucket policies, and ACLs (allow and deny effects).

**Figure 2-47** Matrix of the IAM policies, bucket policies, and ACLs (allow and deny effects)

Bucket Policy	IAM Policy			ACL
	Deny	Allow	Default Deny	
Deny	Deny			Allow
				Default Deny
Allow	Deny	Allow		Allow
				Default Deny
Default Deny		Allow	Deny	Allow
		Deny	Deny	Default Deny

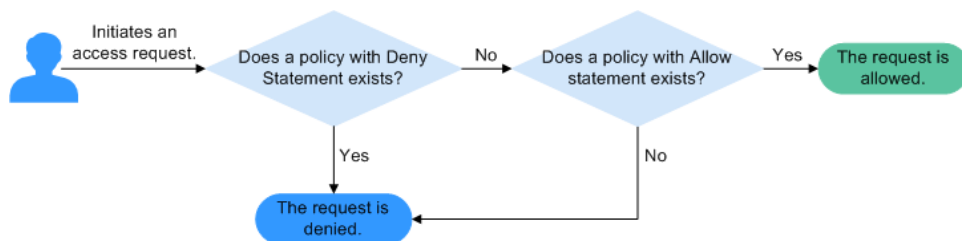
## 2.9.3 Bucket Policy Parameters

### 2.9.3.1 Effect

A bucket policy can either allow or deny requests.

- **Allow:** The policy allows the matched requests.
- **Deny:** The policy denies the matched requests.

When a bucket policy contains both the allow and deny effects, the deny effect prevails. The following figure shows the judgment process.

**Figure 2-48** Determining a bucket policy when the allow and deny statements conflict

1. A user initiates an access request.
2. OBS preferentially searches for bucket policies that have the deny (explicit deny) effect. If a deny statement is found, OBS directly rejects the access. The access request ends.
3. If there is no deny statement, OBS searches for allow statements.
  - If an allow statement is found, OBS allows the access.
  - If no allow statement is found, OBS rejects the access. The access request ends.
4. If an error occurs during the judgment, an error message is generated and returned to the user who initiates the access request.

### 2.9.3.2 Principals

The principals indicate the users bucket policies apply to. These users can be accounts and IAM users. The **Exclude** setting can be used to determine whether the bucket policy applies to the specified principals.

**Specified principals:** By selecting this option (optional), the bucket policy applies to users except the specified ones.

#### NOTE

- **Exclude** not selected: The bucket policy applies to the specified users.
- **Exclude** selected: The bucket policy applies to users except the specified ones.

### 2.9.3.3 Resources

You can apply a bucket policy to the following resources: an entire bucket (including the objects in it), the current bucket, and specified objects in a bucket.

The **Exclude** setting can be used to determine whether the bucket policy applies to the specified resources.

Selecting **Specified resources** for **Exclude** will let the bucket policy apply to the resources except the specified ones.

#### NOTE

If you do not select **Specified resources** for **Exclude**, the bucket policy applies to the specified resources.

## Applying a Bucket Policy to the Entire Bucket (Including the Objects in It)

If you apply the bucket policy to the entire bucket (including the objects in it), actions related to the bucket and objects must be configured in the policy.

## Applying a Bucket Policy to a Bucket

To specify the current bucket as the resource, select **Current bucket**. When configuring actions for the policy, select bucket related actions.

## Applying a Bucket Policy to Specified Objects

To apply the bucket policy to specified objects in a bucket, object-related actions must be configured in the policy. Specifically, select **Specified objects** for **Resources**. The configuration format is as follows:

- For an object, enter the object name (including its folder name if any). If you want to specify the **example.jpg** file in the **imgs-folder** folder in the bucket, enter the following content in the resource text box:

**imgs-folder/example.jpg**

- For an object set, the wildcard asterisk (\*) should be used. The asterisk \* indicates an empty string or any combination of multiple characters. The format rules are as follows:
  - Use only one asterisk (\*) to indicate all objects in a bucket.
  - Use *Object name prefix\** to indicate objects starting with this prefix in a bucket. For example,  
imgs\*
  - Use *\*Object name suffix* to indicate objects ending with this suffix in a bucket. For example,  
\*.jpg

### NOTE

Use commas (,) to separate one object (or object set) from another.

## 2.9.3.4 ActionsActions

The **Exclude** setting can be used to determine whether the bucket policy applies to the specified actions.

Selecting **Specified actions** for **Exclude** will let the bucket policy apply to the actions except the specified ones.

### NOTE

- If you do not select **Specified actions** for **Exclude**, the bucket policy applies to the specified actions.
- By default, **Specified actions** is selected for **Exclude** in the bucket read/write template only. The action exclusion setting in bucket policy templates cannot be modified.

## Actions Related to Buckets

**Table 2-19** Actions related to buckets

Type	Value	Description
General	*	The value supports a wildcard character (*) that indicates all operations can be performed.
	Get*	The value supports a wildcard character (*) that indicates all GET operations can be performed.
	Put*	The value supports a wildcard character (*) that indicates all PUT operations can be performed.
	List*	The value supports a wildcard character (*) that indicates all LIST operations can be performed.
Bucket	HeadBucket	Checks whether a bucket exists and obtains the bucket metadata.
	DeleteBucket	Deletes a bucket.
	GetBucketStorage	Obtains bucket storage information.
	ListBucket	Lists objects in a bucket, and obtains the bucket metadata.
	ListBucketVersions	Lists versioned objects in a bucket.
	ListBucketMultipartUploads	Lists multipart uploads.
	GetBucketAcl	Obtains the bucket ACL information.
	PutBucketAcl	Configures a bucket ACL.
	GetBucketCORS	Obtains the CORS configuration of the bucket.
	PutBucketCORS	Configures CORS for a bucket.
	GetBucketVersioning	Obtains the bucket versioning information.
	PutBucketVersioning	Configures versioning for a bucket.
	GetBucketLocation	Obtains the bucket location.
	GetBucketLogging	Obtains the bucket logging information.
PutBucketLogging	Configures logging for a bucket.	

Type	Value	Description
	GetBucketWebsite	Obtains the static website configuration of the bucket.
	PutBucketWebsite	Configures static website hosting for a bucket.
	DeleteBucketWebsite	Deletes the static website hosting configuration of the bucket.
	GetLifecycleConfigura- tion	Obtains the lifecycle rules of the bucket.
	PutLifecycleConfigura- tion	Configures a lifecycle rule for a bucket.
	GetBucketInventory- Configuration	Obtains the inventory configuration of a bucket.
	PutBucketInventory- Configuration	Configures inventories for a bucket.
	DeleteBucketInventor- yConfiguration	Deletes the inventory configuration of a bucket.
	PutBucketPolicy	Configures a bucket policy.
	GetBucketPolicy	Obtains a bucket policy.
	DeleteBucketPolicy	Deletes a bucket policy.
	PutBucketNotification	Configures event notifications for a bucket.
	GetBucketNotification	Obtains the event notification configuration of a bucket.
	PutBucketStoragePoli- cy	Configures the default storage class for a bucket.
	GetBucketStoragePoli- cy	Obtains the default storage class of a bucket.
	PutReplicationConfi- guration	Configures cross-region replication for a bucket.
	GetReplicationConfi- guration	Obtains the cross-region replication configuration of a bucket.
	DeleteReplicationCon- figuration	Deletes the cross-region replication configuration of a bucket.
	PutBucketTagging	Configures tags for a bucket.
	GetBucketTagging	Obtains bucket tags.
	DeleteBucketTagging	Deletes bucket tags.

Type	Value	Description
	PutBucketQuota	Configures bucket storage quota.
	GetBucketQuota	Queries bucket storage quota.
	PutBucketCustomDomainConfiguration	Binds a user-defined domain name to a bucket.
	GetBucketCustomDomainConfiguration	Obtains the user-defined domain name bound to a bucket.
	DeleteBucketCustomDomainConfiguration	Unbinds a user-defined domain name from a bucket.
	PutDirectColdAccessConfiguration	Configures direct reading for a bucket.
	GetDirectColdAccessConfiguration	Obtains the direct reading configuration of a bucket.
	DeleteDirectColdAccessConfiguration	Deletes the direct reading configuration of a bucket.
	GetEncryptionConfiguration	Obtains the encryption configuration of a bucket.
	PutEncryptionConfiguration	Configures default encryption for a bucket.

## Actions Related to Objects

**Table 2-20** Actions related to objects

Type	Value	Description
General	*	The value supports a wildcard character (*) that indicates all operations can be performed.
	Get*	The value supports a wildcard character (*) that indicates all GET operations can be performed.
	Put*	The value supports a wildcard character (*) that indicates all PUT operations can be performed.
	List*	The value supports a wildcard character (*) that indicates all LIST operations can be performed.
Object	GetObject	Obtains an object and its metadata.

Type	Value	Description
	GetObjectVersion	Obtains the object of a specified version and its metadata.
	PutObject	Performs PUT upload, POST upload, multipart upload, initialization of uploaded parts, and merging of parts.
	GetObjectAcl	Obtains the object ACL information.
	GetObjectVersionAcl	Obtains the ACL information of a specified object version.
	PutObjectAcl	Configures an object ACL.
	PutObjectVersionAcl	Configures the ACL for a specified object version.
	DeleteObject	Deletes an object.
	DeleteObjectVersion	Deletes a specified object version.
	ListMultipartUploadParts	Lists uploaded parts.
	AbortMultipartUpload	Cancels a multipart upload task.
	ModifyObjectMetadata	Modifies object metadata
	RestoreObject	Restores Cold objects.

### 2.9.3.5 Conditions

In addition to effect, principals, resources, and actions, you can specify conditions for a bucket policy. A bucket policy takes effect only when its condition expressions match values contained in the request. **Conditions** is an optional parameter. You can determine whether to use this parameter based on service requirements.

For example, if account **A** needs to be granted with full control permissions for an object uploaded by account **B** in bucket **example**, you can specify that the upload request must contain the **acl** key and set the policy effect to **Allow** for account **A**. The complete condition expression is as follows:

Condition Operator	Key	Value
StringEquals	acl	bucket-owner-full-control

A condition consists of three parts: condition operator, key, and value. Condition operators and keys are associated with each other. For example:



- If a string type condition operator is selected, such as **StringEquals**, the key can only be of the string type, such as **UserAgent**.
- If a date type key is selected, such as **CurrentTime**, the condition operator can only be of the date type, such as **DateEquals**.

**Table 2-21** describes the predefined condition operators provided by OBS.

**Table 2-21** Condition operators

Type	Key	Description
String	StringEquals	Strict matching. Short version: streq
	StringNotEquals	Strict negated matching. Short version: strneq
	StringEqualsIgnoreCase	Strict matching, ignoring case. Short version: streqi
	StringNotEqualsIgnoreCase	Strict negated matching, ignoring case. Short version: strneqi
	StringLike	Loose case-sensitive matching. The values can include a multi-character match wildcard (*) or a single-character match wildcard (?) anywhere in the string. Short version: strl
	StringNotLike	Negated loose case-sensitive matching. The values can include a multi-character match wildcard (*) or a single-character match wildcard (?) anywhere in the string. Short version: strnl
Numeric	NumericEquals	Strict matching. Short version: numeq
	NumericNotEquals	Strict negated matching. Short version: numneq
	NumericLessThan	"Less than" matching. Short version: numlt
	NumericLessThanEquals	"Less than or equals" matching. Short version: numlteq
	NumericGreaterThan	"Greater than" matching. Short version: numgt
	NumericGreaterThanEquals	"Greater than or equals" matching. Short version: numgteq
Date	DateEquals	Strict matching. Short version: dateeq

Type	Key	Description
	DateNotEquals	Strict negated matching. Short version: dateneq
	DateLessThan	Indicates that the date is earlier than a specific date. Short version: datelt
	DateLessThanEquals	Indicates that the date is earlier than or equal to a specific date. Short version: datelteq
	DateGreaterThan	Indicates that the date is later than a specific date. Short version: dategt
	DateGreaterThanEquals	Indicates that the date is later than or equal to a specific date. Short version: dategteq
Boolean	Bool	Strict Boolean matching
IP address	IpAddress	Takes effect only on a specified IP address or IP address range. Example: <b>x.x.x.x/24</b>
	NotIpAddress	Takes effect only on all except the specified IP address or IP address range. Example: <b>x.x.x.x/24</b>

A condition can contain any of the three types of keys: general keys, keys related to bucket actions, and keys related to object actions.

**Table 2-22** General keys

Key	Type	Description
CurrentTime	Date	Indicates the date when the request is received by the server. The date format must comply with ISO 8601.
EpochTime	Numeric	Indicates the time when the request is received by the server, which is expressed as seconds since 1970.01.01 00:00:00 UTC, regardless of the leap seconds.
SecureTransport	Bool	Requests whether to use SSL.
SourceIp	IP address	Source IP address from which the request is sent
UserAgent	String	Requested client software agent

Key	Type	Description
Referer	String	Indicates the link from which the request is sent.

**Table 2-23** Keys related to bucket actions

Action	Optional Key	Description	Description
ListBucket	prefix	Type: String. Lists objects that begin with the specified prefix.	If <b>prefix</b> , <b>delimiter</b> , and <b>max-keys</b> are configured, the key-value pair meeting the conditions must be specified in the List operation for the bucket policy to take effect.  For example, if a bucket policy (with the conditional operator set to <b>NumericEquals</b> , the key to <b>max-keys</b> , and the value to <b>100</b> ) that allows anonymous users to read data is configured for a bucket, the anonymous users must add <b>?max-keys=100</b> to the end of the bucket domain name for listing objects. The listed objects are the first 100 objects in alphabetic order.
	max-keys	Type: Numeric. Sets the maximum number of objects. Returned objects are listed in alphabetic order.	
ListBucketVersions	prefix	Type: String. Lists multi-version objects whose name starts with the specified prefix.	
	max-keys	Type: Numeric. Sets the maximum number of objects. Returned objects are listed in alphabetic order.	

Action	Optional Key	Description	Description
PutBucketAcl	acl	Type: String. Configures the bucket ACL. When modifying a bucket ACL, you can use the request that contains a canned ACL setting in its header. Value options of a canned ACL setting: <b>private public-read public-read-write authenticated-read bucket-owner-read bucket-owner-full-control log-delivery-write</b>	None

**Table 2-24** Keys related to object actions

Action	Optional Key	Description
PutObject	acl	Type: String. Configures the object ACL. When uploading an object, you can use the request that contains a canned ACL setting in its header. Value options of a canned ACL setting: <b>private public-read public-read-write authenticated-read bucketowner-read bucket-owner-full-control log-delivery-write.</b>
	copysource	Type: String. Specifies names of the source bucket and the source object. Format: <b>/bucketname/keyname</b>
	metadata-directive	Type: String. Specifies whether to copy the metadata from the source object or replace with the metadata in the request. Values: <b>COPY REPLACE</b>
PutObjectAcl	acl	Type: String. Configures the object ACL. When uploading an object, you can use the request that contains a canned ACL setting in its header. Value options of a canned ACL setting: <b>private public-read public-read-write authenticated-read bucketowner-read bucket-owner-full-control log-delivery-write.</b>
GetObjectVersion	VersionId	Type: String. Obtains the object with the specified version ID.

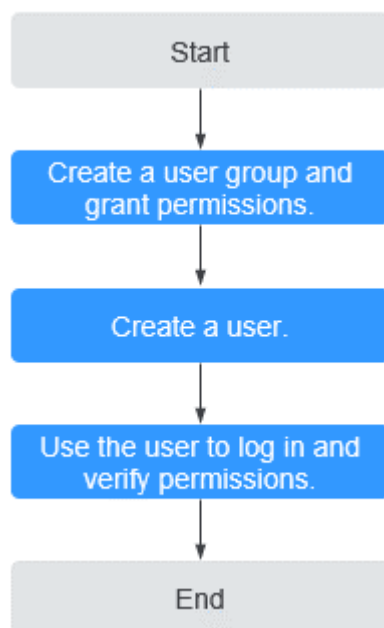
Action	Optional Key	Description
GetObjectVersionAcl	VersionId	Type: String. Obtains the ACL of the object with specified version ID.
PutObjectVersionAcl	VersionId	Type: String. Specifies a version ID.
	acl	Type: String. Configures the ACL of the object with the specified version ID. When uploading an object, you can use the request that contains a canned ACL setting in its header. Value options of a canned ACL setting: <b>private public-read public-read-write authenticated-read bucketowner-read bucket-owner-full-control log-delivery-write.</b>
DeleteObjectVersion	VersionId	Type: String. Deletes the object with the specified version ID.

## 2.9.4 Configuring IAM Policies

### 2.9.4.1 Creating an IAM User and Granting OBS Permissions

#### Process

Figure 2-49 Process of granting an IAM user the OBS permissions



## Procedure

**Step 1** Log in to the management console with your account.

**Step 2** On the top menu bar, choose **Service List > Management & Deployment > Identity and Access Management**. The IAM console is displayed.

**Step 3** Create a user group and assign OBS permissions to it.

A user group is a collection of users. By assigning permissions to a user group, you assign permissions to the users in this group. After you create an IAM user, add it to one or more user groups, so that it can inherit the permissions from the groups.

1. In the navigation pane, choose **User Groups**. The **User Groups** page is displayed.

2. Click **Create User Group**.

3. Enter a user group name and click **OK**.

The user group is displayed in the user group list once the creation is complete.

4. Locate the user group you created and click **Authorize** in the **Operation** column of the row.

5. Under **Select Policy/Role**, filter policies based on policy types in the upper right corner, required policy names, and click **Next**.

6. Under **Select Scope**, select **Global services** and click **OK**.

### NOTE

In the policy content area, you can view the authorization details.

Due to data caching, an RBAC policy or a fine-grained policy involving OBS actions will take effect 10 to 15 minutes after it is attached to a user, an enterprise project, or a user group.

**Step 4** Create a user. For details, see [Creating an IAM user](#).

**Step 5** Use the created IAM user to log in to OBS Console and verify the user permissions.

----End

### 2.9.4.2 Configuring Fine-Grained Policies

Custom policies can be created to supplement the system-defined policies of OBS. For the actions supported for custom policies, see [Bucket-Related Actions](#) and [Object-Related Actions](#).

You can create custom policies in either of the following two ways:

- Visual editor: Select cloud services, actions, resources, and request conditions without the need to know policy syntax.
- JSON: Edit JSON policies from scratch or based on an existing policy.

For details, see [Creating a Custom Policy](#). The following provides examples of common OBS custom policies.

## Example Custom Policies

- Example 1: Grant users all OBS permissions.

This policy allows users to perform any operation on OBS.

```
{
  "Version": "1.1",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "obs:*"
      ]
    }
  ]
}
```

- Example 2: Grant users all OBS Console permissions.

This policy allows users to perform all operations on OBS Console.

When a user logs in to OBS Console, the user may access resources of other services such as audit information in CTS. Therefore, in addition to the OBS permissions in example 1, you also need to configure the access permissions to other services. You need to configure the **Tenant Guest** permission for the global project and regional projects based on the services and regions that you use.

```
{
  "Version": "1.1",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "obs:*"
      ]
    }
  ]
}
```

- Example 3: Grant users the read-only permission for all directories in a bucket.

This policy allows users to list and download all objects in bucket **obs-example**.

```
{
  "Version": "1.1",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "obs:object:GetObject",
        "obs:bucket:ListBucket"
      ],
      "Resource": [
        "obs:*:object:obs-example/*",
        "obs:*:bucket:obs-example"
      ]
    }
  ]
}
```

- Example 4: Grant users the read-only permission for a specified directory in a bucket.

This policy allows users to download objects in only the **my-project/** directory of bucket **obs-example**. Objects in other directories can be listed but cannot be downloaded.

```
{
  "Version": "1.1",
  "Statement": [
```

```
{
  "Effect": "Allow",
  "Action": [
    "obs:object:GetObject",
    "obs:bucket:ListBucket"
  ],
  "Resource": [
    "obs:*:object:obs-example/my-project/*",
    "obs:*:bucket:obs-example"
  ]
}
```

- Example 5: Grant users the read/write permissions for a specified directory in a bucket.

This policy allows users to list, download, upload, and delete objects in the **my-project** directory of bucket **obs-example**.

```
{
  "Version": "1.1",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "obs:object:GetObject",
        "obs:object:ListMultipartUploadParts",
        "obs:bucket:ListBucket",
        "obs:object:DeleteObject",
        "obs:object:PutObject"
      ],
      "Resource": [
        "obs:*:object:obs-example/my-project/*",
        "obs:*:bucket:obs-example"
      ]
    }
  ]
}
```

- Example 6: Grant users all permissions for a bucket.

This policy allows users to perform any operation on bucket **obs-example**.

```
{
  "Version": "1.1",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "obs:*"
      ],
      "Resource": [
        "obs:*:bucket:obs-example",
        "obs:*:object:obs-example/*"
      ]
    }
  ]
}
```

- Example 7: Grant users the permission to deny object upload.

A deny policy must be used together with other policies. If the permissions assigned to a user contain both "Allow" and "Deny", the "Deny" permissions take precedence over the "Allow" permissions.

If you grant the system policy OBS Operator to a user but do not want the user to have the object upload permission (which is also a permission allowed by OBS Operator), you can create a custom policy besides the OBS Operator policy, to deny the user's upload permission. According to the authorization principle, the policy with the deny statement takes precedence, so that the



user can perform all operations allowed by OBS Operator, except uploading objects. The following is an example of a deny policy:

```
{
  "Version": "1.1",
  "Statement": [
    {
      "Effect": "Deny",
      "Action": [
        "obs:object:PutObject"
      ]
    }
  ]
}
```

### 2.9.4.3 OBS Resources

A resource is an object that exists within a service. OBS resources include buckets and objects. You can select these resources by specifying their paths.

**Table 2-25** OBS resources and their paths

Resource Type	Resource Name	Path
Buckets	Bucket	[Format] <b>obs:*:*:bucket:</b> <i>Bucket name</i> [Notes] IAM automatically generates the prefix <b>obs:*:*:bucket:</b> for bucket resource paths. By adding <i>Bucket name</i> to the end of the generated prefix, you can define a specific path. An asterisk * is allowed to indicate any bucket. An example is given as follows: <b>obs:*:*:bucket:*</b>
Objects	Object	[Format] <b>obs:*:*:object:</b> <i>Bucket name/ Object name</i> [Notes] IAM automatically generates the prefix <b>obs:*:*:object:</b> for object resource paths. By adding <i>Bucket name/Object name</i> to the end of the generated prefix, you can define a specific path. An asterisk * is allowed to any object in the bucket. An example is given as follows: <b>obs:*:*:object:my-bucket/my-object/*</b> (indicating any object in the <b>my-object</b> directory of bucket <b>my-bucket</b> )

## 2.9.5 Configuring a Bucket Policy

### 2.9.5.1 Creating a Bucket Policy with a Template

OBS Console provides bucket policy templates for eight typical scenarios. You can use these templates to quickly configure bucket policies.

#### Procedure

- Step 1** In the bucket list, click the bucket you want to operate to go to the **Objects** page.
- Step 2** In the navigation pane, choose **Permissions > Bucket Policies**.
- Step 3** Click **Create**.
- Step 4** Choose a policy template. For details about the parameters, see [Bucket Policies and Object Policies](#).

**Figure 2-50** Choosing the Public Read template

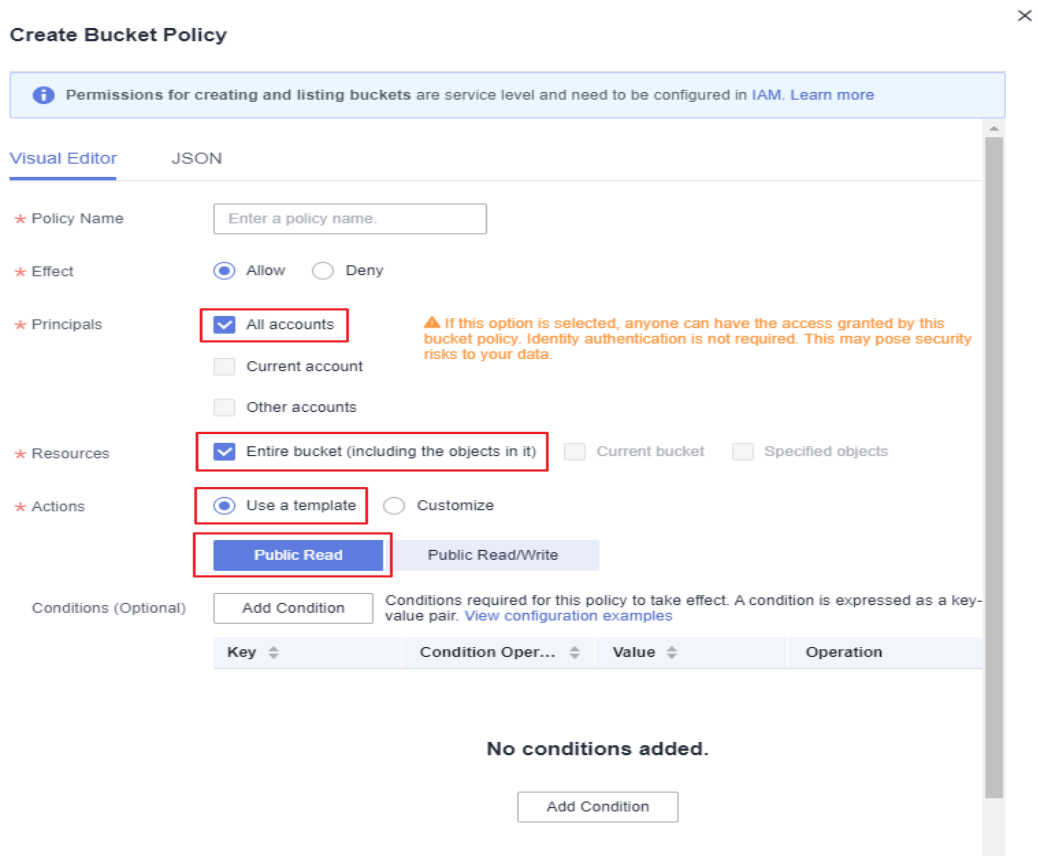


Figure 2-51 Choosing the Public Read/Write template

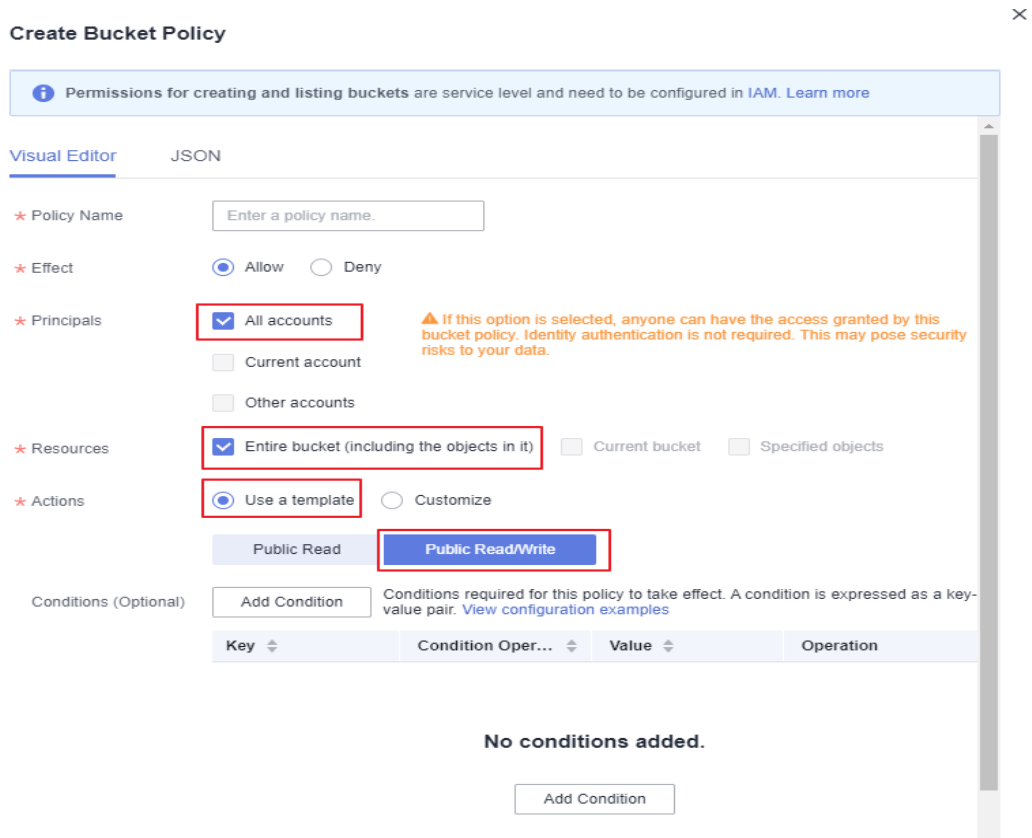


Figure 2-52 Choosing the Bucket Read-Only template

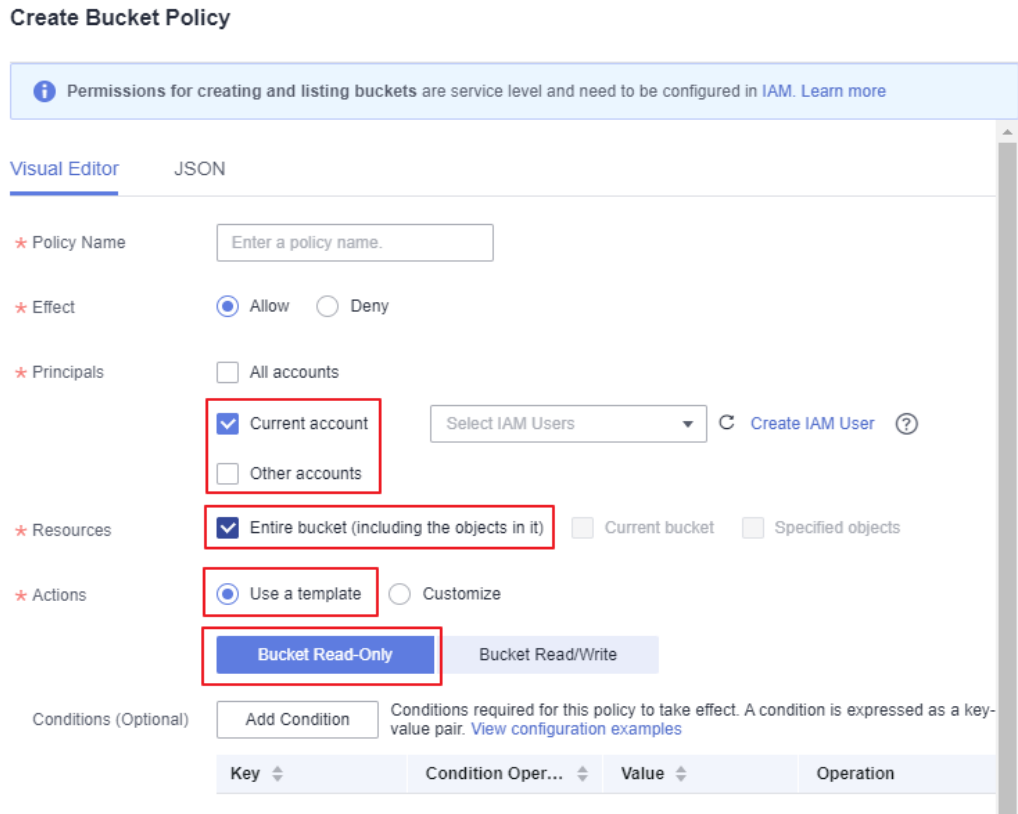


Figure 2-53 Choosing the Bucket Read/Write template

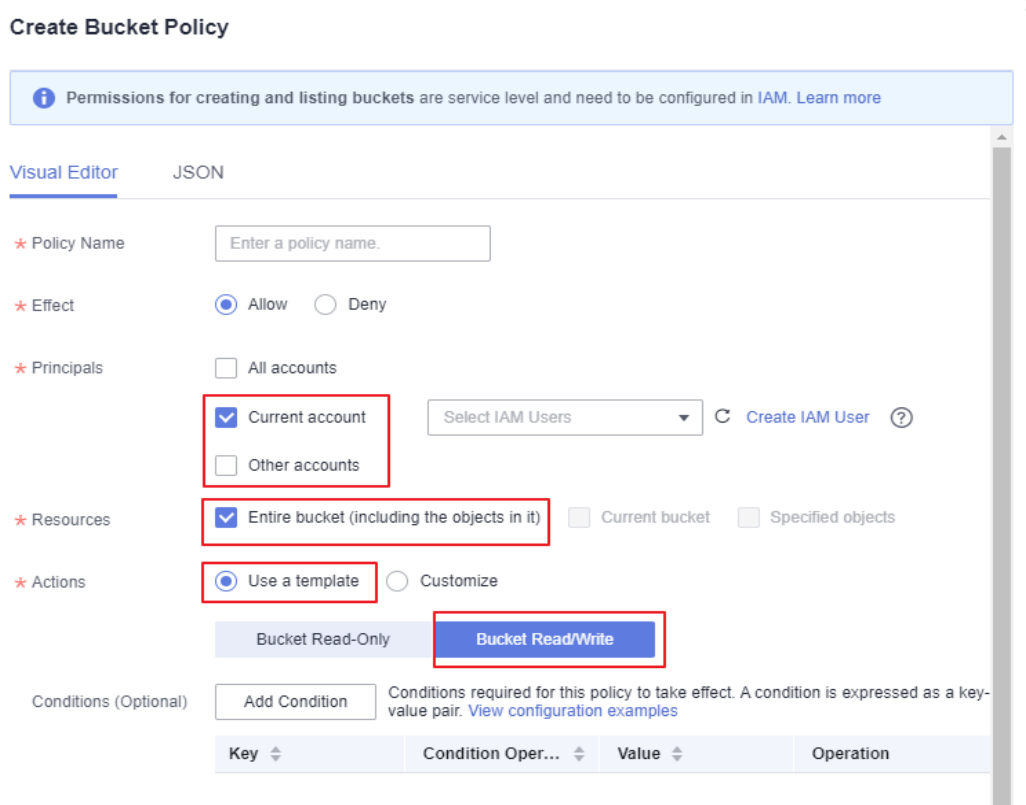


Figure 2-54 Choosing the Directory Read-Only template

**Create Bucket Policy**

Permissions for creating and listing buckets are service level and need to be configured in IAM. [Learn more](#)

Visual Editor JSON

\* Policy Name

\* Effect  Allow  Deny

\* Principals

- All accounts ⚠ If this option is selected, anyone can have the access granted by this bucket policy. Identity authentication is not required. This may pose security risks to your data.
- Current account
- Other accounts

\* Resources

- Entire bucket (including the objects in it)
- Current bucket
- Specified objects

Bucket selected:

Format: Folder name/Object name, for example, testdir/a.txt. \* indicates all objects.

[Add](#)

\* Actions

- Use a template
- Customize

- Directory Read-Only**
- Directory Read/Write

Conditions (Optional)  Conditions required for this policy to take effect. A condition is expressed as a key-value pair. [View configuration examples](#)

Key	Condition Oper...	Value	Operation
-----	-------------------	-------	-----------

Figure 2-55 Choosing the Directory Read/Write template

### Create Bucket Policy

**Permissions for creating and listing buckets are service level and need to be configured in IAM. [Learn more](#)**

**Visual Editor**    JSON

\* Policy Name

\* Effect  Allow     Deny

\* Principals

- All accounts **▲ If this option is selected, anyone can have the access granted by this bucket policy. Identity authentication is not required. This may pose security risks to your data.**
- Current account
- Other accounts

\* Resources

- Entire bucket (including the objects in it)
- Current bucket
- Specified objects

Bucket selected:

;

Format: Folder name/Object name, for example, testdir/a.txt. \* indicates all objects.

Add

\* Actions

- Use a template
- Customize

Directory Read-Only    **Directory Read/Write**

Conditions (Optional)  Conditions required for this policy to take effect. A condition is expressed as a key-value pair. [View configuration examples](#)

Key	Condition Oper...	Value	Operation
-----	-------------------	-------	-----------

**Figure 2-56** Choosing the Object Read-Only template

**Create Bucket Policy**

**Permissions for creating and listing buckets are service level and need to be configured in IAM. [Learn more](#)**

Visual Editor    JSON

\* Policy Name

\* Effect  Allow  Deny

\* Principals

- All accounts **▲ If this option is selected, anyone can have the access granted by this bucket policy. Identity authentication is not required. This may pose security risks to your data.**
- Current account
- Other accounts

\* Resources

- Entire bucket (including the objects in it)
- Current bucket
- Specified objects

Format: Folder name/Object name, for example, testdir/a.txt. \* indicates all objects.

⊕ Add

\* Actions

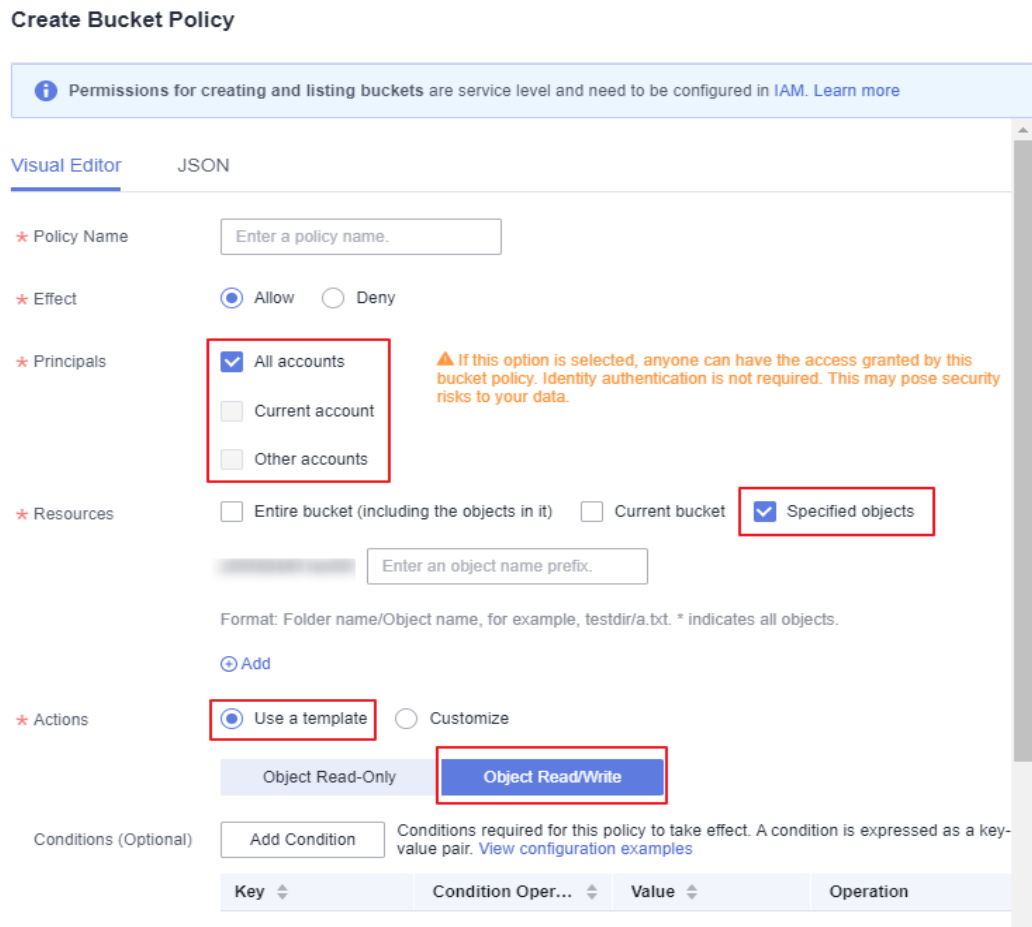
- Use a template
- Customize

**Object Read-Only**    Object Read/Write

Conditions (Optional)  Conditions required for this policy to take effect. A condition is expressed as a key-value pair. [View configuration examples](#)

Key	Condition Oper...	Value	Operation
-----	-------------------	-------	-----------

**Figure 2-57** Choosing the Object Read/Write template



**Table 2-26** Bucket policy templates

Principal	Resource	Template	Actions Allowed	Advanced Settings
All accounts	Entire bucket (including the objects in it)	Public Read See <a href="#">Figure 2-50</a> .	<p><b>Allows anonymous users to perform the following actions on a bucket and the objects in it:</b></p> <ul style="list-style-type: none"> <li>HeadBucket (to check whether the bucket exists and obtain the bucket metadata)</li> <li>GetBucketLocation (to get the bucket location)</li> <li>GetObject (to obtain object content and metadata)</li> <li>RestoreObject (to restore objects from Cold storage)</li> <li>GetObjectVersion (to obtain the content and metadata of a specified object version)</li> </ul>	Excluding the specified actions is not allowed.



Principal	Resource	Template	Actions Allowed	Advanced Settings
		Public Read/Write  See <a href="#">Figure 2-51</a> .	<p><b>Allows anonymous users to perform the following actions on a bucket and the objects in it:</b></p> <ul style="list-style-type: none"> <li>ListBucket (to list objects in the bucket and obtain the bucket metadata)</li> <li>ListBucketVersions (to list object versions in the bucket)</li> <li>HeadBucket (to check whether the bucket exists and obtain the bucket metadata)</li> <li>GetBucketLocation (to get the bucket location)</li> <li>PutObject (to upload objects using PUT and POST, upload parts, initiate multipart uploads, and assemble parts)</li> <li>GetObject (to obtain object content and metadata)</li> <li>ModifyObjectMetaData (to modify object metadata)</li> <li>ListBucketMultipartUploads (to list multipart uploads)</li> <li>ListMultipartUploadParts (to list uploaded parts)</li> <li>AbortMultipartUpload (to abort multipart uploads)</li> <li>RestoreObject (to restore objects from Cold storage)</li> <li>GetObjectVersion (to obtain the content and metadata of a specified object version)</li> <li>PutObjectAcl (to configure the object ACL)</li> <li>GetObjectVersionAcl (to obtain the ACL of a specified object version)</li> <li>GetObjectAcl (to obtain the object ACL)</li> </ul>	Excluding the specified actions is not allowed.

Principal	Resource	Template	Actions Allowed	Advanced Settings
Current account/ Other accounts/ Delegated accounts	Entire bucket (including the objects in it)	Bucket Read-Only See <a href="#">Figure 2-52</a> .	<b>Allows specified accounts to perform the following actions on a bucket and the objects in it:</b> Get* (all GET actions) List* (all LIST actions) HeadBucket (to check whether the bucket exists and obtain the bucket metadata)	Excluding the specified actions is not allowed.
		Bucket Read/Write See <a href="#">Figure 2-53</a> .	<b>Allows specified accounts to perform all actions excluding the following ones on a bucket and the objects in it:</b> DeleteBucket (to delete the bucket) PutBucketPolicy (to configure a bucket policy) PutBucketAcl (to configure the bucket ACL)	The specified actions are excluded.

Principal	Resource	Template	Actions Allowed	Advanced Settings
All accounts/ Current account/ Other accounts/ Delegated accounts	Current bucket + Specified objects	Directory Read-Only See <a href="#">Figure 2-54</a> .	<p><b>Allows all accounts or specified accounts to perform the following actions on the current bucket and the specified resources in it:</b></p> <p>GetObject (to obtain object content and metadata)</p> <p>GetObjectVersion (to obtain the content and metadata of a specified object version)</p> <p>GetObjectVersionAcl (to obtain the ACL of a specified object version)</p> <p>GetObjectAcl (to obtain the object ACL)</p> <p>RestoreObject (to restore objects from Cold storage)</p> <p>ListBucket (to list objects in the bucket and obtain the bucket metadata)</p> <p>ListBucketVersions (to list object versions in the bucket)</p> <p>HeadBucket (to check whether the bucket exists and obtain the bucket metadata)</p> <p>GetBucketLocation (to get the bucket location)</p> <p><b>NOTE</b> If you apply the policy to <b>All accounts</b>, <b>ListBucket</b> and <b>ListBucketVersions</b> are not included in the template.</p>	Excluding the specified actions is not allowed.

Principal	Resource	Template	Actions Allowed	Advanced Settings
		Directory Read/Write See <a href="#">Figure 2-55</a> .	<p><b>Allows all accounts or specified accounts to perform the following actions on the current bucket and the specified resources in it:</b></p> <p>PutObject (to upload objects using PUT and POST, upload parts, initiate multipart uploads, and assemble parts)</p> <p>GetObject (to obtain object content and metadata)</p> <p>GetObjectVersion (to obtain the content and metadata of a specified object version)</p> <p>ModifyObjectMetaData (to modify object metadata)</p> <p>ListBucketMultipartUploads (to list multipart uploads)</p> <p>ListMultipartUploadParts (to list uploaded parts)</p> <p>AbortMultipartUpload (to abort multipart uploads)</p> <p>GetObjectVersionAcl (to obtain the ACL of a specified object version)</p> <p>GetObjectAcl (to obtain the object ACL)</p> <p>PutObjectAcl (to configure the object ACL)</p> <p>RestoreObject (to restore objects from Cold storage)</p> <p>ListBucket (to list objects in the bucket and obtain the bucket metadata)</p> <p>ListBucketVersions (to list object versions in the bucket)</p> <p>HeadBucket (to check whether the bucket exists and obtain the bucket metadata)</p> <p>GetBucketLocation (to get the bucket location)</p>	Excluding the specified actions is not allowed.

Principal	Resource	Template	Actions Allowed	Advanced Settings
All accounts/ Current account/ Other accounts/ Delegated accounts	Specified objects	Object Read-Only See <a href="#">Figure 2-56</a> .	<p><b>Allows all accounts or specified accounts to perform the following actions on specified resources in the bucket:</b></p> <ul style="list-style-type: none"> <li>GetObject (to obtain object content and metadata)</li> <li>GetObjectVersion (to obtain the content and metadata of a specified object version)</li> <li>GetObjectVersionAcl (to obtain the ACL of a specified object version)</li> <li>GetObjectAcl (to obtain the object ACL)</li> <li>RestoreObject (to restore objects from Cold storage)</li> </ul>	Excluding the specified actions is not allowed.
		Object Read/Write See <a href="#">Figure 2-57</a> .	<p><b>Allows all accounts or specified accounts to perform the following actions on specified resources in the bucket:</b></p> <ul style="list-style-type: none"> <li>PutObject (to upload objects using PUT and POST, upload parts, initiate multipart uploads, and assemble parts)</li> <li>GetObject (to obtain object content and metadata)</li> <li>GetObjectVersion (to obtain the content and metadata of a specified object version)</li> <li>ModifyObjectMetaData (to modify object metadata)</li> <li>ListMultipartUploadParts (to list uploaded parts)</li> <li>AbortMultipartUpload (to abort multipart uploads)</li> <li>GetObjectVersionAcl (to obtain the ACL of an object version)</li> <li>GetObjectAcl (to obtain the object ACL)</li> <li>PutObjectAcl (to configure the object ACL)</li> <li>RestoreObject (to restore objects from Cold storage)</li> </ul>	Excluding the specified actions is not allowed.

**Step 5** Complete the bucket policy configuration.

Some bucket policy templates require a configuration of principals or resources. You can also change the existing settings of a template, including the policy name, principals, resources, actions, and conditions. For details, see [Bucket Policy Parameters](#).

**Step 6** Click **Create** in the lower right corner.

----End

### 2.9.5.2 Creating a Custom Bucket Policy (Visual Editor)

You can customize bucket policies based on your needs. A custom bucket policy consists of five basic elements: effect, principals, resources, actions, and conditions.

#### Procedure

**Step 1** In the bucket list, click the bucket you want to operate to go to the **Objects** page.

**Step 2** In the navigation pane, choose **Permissions > Bucket Policies**.

**Step 3** Click **Create**.

**Step 4** Configure a bucket policy.

**Figure 2-58** Configuring a bucket policy

**Create Bucket Policy** [Learn more](#) ×

**Permissions for creating and listing buckets are service level and need to be configured in IAM. [Learn more](#)**

**Visual Editor**    JSON

\* Policy Name:

\* Effect:  Allow     Deny

\* Principal:  All accounts     Current account     Other accounts  
⚠ If this option is selected, anyone can have the access granted by this bucket policy. Identity authentication is not required. This may pose security risks to your data.

\* Resources:  Entire bucket (including the objects in it)     Current bucket     Specified objects

\* Actions:  Use a template     **Customize**

--Select--   

Conditions (Optional)        Conditions required for this policy to take effect. A condition is expressed as a key-value pair. [View configuration examples](#)

Key	Condition Oper...	Value	Operation
No conditions added.			

**Table 2-27** Parameters for configuring a custom bucket policy

Parameter		Description
Method		Visual editor or JSON. The visual editor is used here. For details about configurations in the JSON view, see <a href="#">Creating a Custom Bucket Policy (JSON View)</a> .
Policy Name		Enter a bucket policy name.
Policy content	Effect	<ul style="list-style-type: none"> <li>• <b>Allow</b>: The policy allows the matched requests.</li> <li>• <b>Deny</b>: The policy denies the matched requests.</li> </ul>

Parameter		Description
	Principals	<ul style="list-style-type: none"> <li>• <b>All accounts:</b> The bucket policy applies to anonymous users.</li> <li>• <b>Current account:</b> The bucket policy applies to one or more IAM users specified under the current account.</li> <li>• <b>Other accounts:</b> The bucket policy applies to one or more accounts specified.</li> </ul> <p><b>NOTE</b> You can obtain the account ID and IAM user ID from the <b>My Credentials</b> page.</p> <p>Accounts should be configured in the <i>Domain ID/IAM user ID</i> format, with each one on a separate line.</p> <p><i>Account ID/*</i> indicates that permission is granted to all IAM users under the account.</p> <ul style="list-style-type: none"> <li>• <b>Delegated accounts:</b> Delegated accounts can be added only after <b>Other accounts</b> is selected.</li> </ul> <p><b>NOTE</b> Delegated accounts should be configured in the <i>ID of a delegating account/Agency name</i> format. Multiple delegated accounts are allowed, with each one on a separate line.</p>
	Resources	<ul style="list-style-type: none"> <li>• <b>Entire bucket (including the objects in it):</b> The policy applies to the bucket and the objects in it. You can configure bucket and object actions in this policy.</li> <li>• <b>Current bucket:</b> The policy applies to the current bucket. You can configure bucket actions in this policy.</li> <li>• <b>Specified objects:</b> The policy applies to specified objects in the bucket. You can configure object actions in this policy.</li> </ul> <p><b>NOTE</b></p> <ol style="list-style-type: none"> <li>1. Multiple resource paths can be specified.</li> <li>2. A resource path should be configured in the <i>Folder name/Object name</i> format, for example, <b>testdir/a.txt</b>. To specify the <b>testdir</b> folder and all objects in it, enter <b>testdir/*</b>.</li> <li>3. You can specify a specific object, an object set, or a directory. * indicates all objects in the bucket. To specify a specific object, enter the object name.</li> </ol> <p>To specify a set of objects, enter <i>Object name prefix*</i>, <i>*Object name suffix</i>, or <i>*</i>. For example, <b>testdir/*</b> indicates objects in the <b>testdir</b> folder, and <b>testprefix*</b> indicates objects whose prefix is <b>testprefix</b>.</p>



Parameter		Description
	Actions	<ul style="list-style-type: none"> <li>• <b>Actions:</b> Choose <b>Customize</b>.</li> <li>• <b>Select Actions:</b> See <a href="#">Actions</a>.</li> </ul> <p><b>NOTE</b></p> <ol style="list-style-type: none"> <li>1. If you select <b>Entire bucket (including the objects in it)</b> for <b>Resources</b>, common actions, bucket actions, and object actions will be available for you to choose from.</li> <li>2. If you select <b>Current bucket</b> for <b>Resources</b>, common actions and bucket actions will be available for you to choose from.</li> <li>3. If you select <b>Specified objects</b> for <b>Resources</b>, common actions and object actions will be available for you to choose from.</li> <li>4. If you select both <b>Current bucket</b> and <b>Specified objects</b> for <b>Resources</b>, common actions, bucket actions, and object actions will be available for you to choose from.</li> </ol>
	Conditions (Optional)	<ul style="list-style-type: none"> <li>• <b>Key:</b> See <a href="#">Conditions</a>.</li> <li>• <b>Conditional Operator:</b> See <a href="#">Conditions</a>.</li> <li>• <b>Value:</b> The entered value is associated with the key.</li> </ul>
	Advanced Settings > Exclude (Optional)	<ul style="list-style-type: none"> <li>• <b>Specified principals:</b> By selecting this option, the bucket policy applies to users except the specified ones.</li> </ul> <p><b>NOTE</b></p> <ol style="list-style-type: none"> <li>1. If you do not select this option, the bucket policy applies to the specified users.</li> </ol> <ul style="list-style-type: none"> <li>• <b>Specified resources:</b> By selecting this option, the bucket policy applies to resources except the specified ones.</li> </ul> <p><b>NOTE</b></p> <ol style="list-style-type: none"> <li>1. If you do not select this option, the bucket policy applies to the specified resources.</li> </ol> <ul style="list-style-type: none"> <li>• <b>Specified actions:</b> By selecting this option, the bucket policy applies to actions except the specified ones.</li> </ul> <p><b>NOTE</b></p> <ol style="list-style-type: none"> <li>1. If you do not select this option, the bucket policy applies to the specified actions.</li> <li>2. By default, <b>Specified actions</b> is selected for <b>Exclude</b> in the bucket read/write template only. The action exclusion setting in bucket policy templates cannot be modified.</li> </ol>

**Step 5** Click **Create** in the lower right corner.

----End

### 2.9.5.3 Creating a Custom Bucket Policy (JSON View)

If you are familiar with the JSON syntax and OBS bucket policies, you can code a bucket policy in the JSON view. There is no limit on the number of bucket policies (statements) for a bucket, but the total size of JSON descriptions of all bucket policies in a bucket cannot exceed 20 KB.

#### Procedure

- Step 1** In the bucket list, click the bucket you want to operate to go to the **Objects** page.
- Step 2** In the navigation pane, choose **Permissions > Bucket Policies**.
- Step 3** Click **Create** and click the **JSON** tab.
- Step 4** Edit the bucket policy. Below gives a bucket policy example in JSON:

```
{
  "Statement": [
    {
      "Action": [
        "CreateBucket",
        "DeleteBucket"
      ],
      "Effect": "Allow",
      "Principal": {
        "ID": [
          "domain/account ID",
          "domain/account ID:user/User ID"
        ]
      },
      "Condition": {
        "NumericNotEquals": {
          "Referer": "sdf"
        },
        "StringNotLike": {
          "Delimiter": "ouio"
        }
      },
      "Resource": "000-02/key01"
    }
  ]
}
```

**Table 2-28** Parameters for creating a bucket policy in JSON

Parameter	Description
Action	Actions the bucket policy applies to. For details, see <a href="#">Actions</a> .
Effect	Effect of the bucket policy. For details, see <a href="#">Effect</a> .

Parameter	Description
Principal	Users the bucket policy is applied to. You can obtain the user ID on the <b>My Credentials</b> page by logging in to the console as the user to be authorized. Principals should be configured as follows: <ul style="list-style-type: none"><li>• <b>domain/Account ID</b> (indicating that the principal is an account)</li><li>• <b>domain/Account ID:user/User ID</b> (indicating that the principal is a user under an account)</li></ul>
Condition	Conditions under which the bucket policy takes effect. For details, see <a href="#">Conditions</a> .
Resource	Resources the bucket policy is applied to. For details, see <a href="#">Resources</a> .

**Step 5** Click **Create**.

----End

## 2.9.6 Configuring an Object Policy

Object policies are applied to the objects in a bucket. With an object policy, you can configure conditions and actions for objects in a bucket.

### Procedure

**Step 1** In the bucket list, click the bucket you want to operate to go to the **Objects** page.

**Step 2** In the row containing the object for which you want to configure a policy, choose **More > Configure Object Policy** in the **Operation** column. The **Configure Object Policy** page is displayed.

You can customize a policy or use a preset template to configure one as needed.

- **Using a preset template:** The system presets object policy templates for two typical scenarios. You can use the templates to quickly configure object policies.
- **Customizing a policy:** You can also customize an object policy based on your needs. A custom object policy consists of five basic elements: effect, principals, resources, actions, and conditions, similar to a bucket policy. For details, see [Bucket Policy Parameters](#). The resource is the selected object and is automatically configured by the system. For details about how to customize an object policy, see [Creating a Custom Bucket Policy \(Visual Editor\)](#). Different from customizing a bucket policy, to customize an object policy, you:
  - a. Do not need to specify the resource.

- b. Can configure only object-related actions.

----End

## 2.9.7 Configuring a Bucket ACL

### Prerequisites

You are the bucket owner or you have the permission to write the bucket ACL.

### Procedure

**Step 1** In the bucket list, click the bucket you want to operate to go to the **Objects** page.

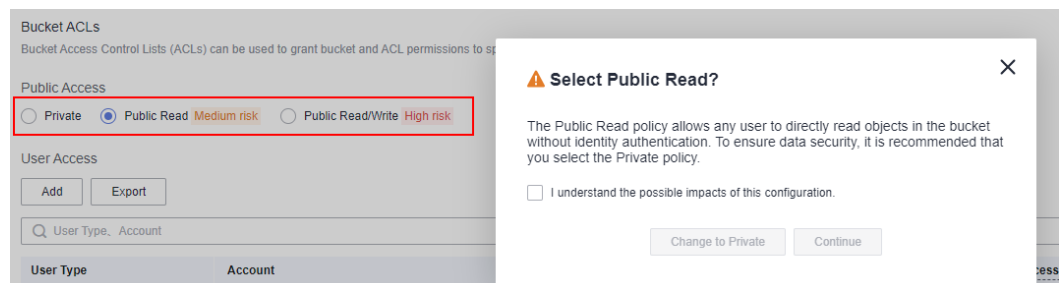
**Step 2** In the navigation pane, choose **Permissions > Bucket ACLs**.

**Step 3** On the **Bucket ACLs** page, choose a permission from **Private**, **Public Read**, and **Public Read/Write** to grant bucket ACL permission for anonymous users.

#### NOTE

1. After you change **Public Read** or **Public Read/Write** to **Private**, only the bucket owner or object owner has the access.
2. After you change **Private** to **Public Read**, anyone can read objects in the bucket. No identity authentication is required.
3. After you change **Private** to **Public Read/Write**, anyone can read, write, and delete objects in the bucket. No identity authentication is required.

**Figure 2-59** Changing a public access permission



**Step 4** In the **Operation** column, click **Edit** to grant the owner, anonymous user, or log delivery user required ACL permissions for the bucket.

**Step 5** In the middle of the page, click **Export** to get the bucket ACL configuration. The file includes the user type, account, bucket access, and ACL access.

**Step 6** In the middle of the page, click **Add** to apply specific ACL permissions to an account.

Enter an account ID or account name and specify ACL permissions for the account. You can obtain the account ID or account name from the **My Credentials** page.

Click **OK**.

#### NOTE

To select **Object read** for **Object Permission**, you must select **Read** for **Access to Bucket**.

Figure 2-60 Granting permissions

×

### Add Account Authorization

Account

ACLs are configured for accounts but not IAM users here. [View relationship between an account and its IAM users](#)

⚠ Only an account ID is supported.

Access to Bucket  Read  Write

Object Permission  Read

Access to ACL  Read  Write

----End

## 2.9.8 Configuring an Object ACL

### Prerequisites

You are the object owner or you have the permission to write the object ACL.

An object owner is the account that uploads the object, but may not be the owner of the bucket that stores the object. 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, account **B**, instead of the bucket owner account **A**, is the owner of the object. By default, account **A** is not allowed to access this object and cannot read or modify the object ACL.

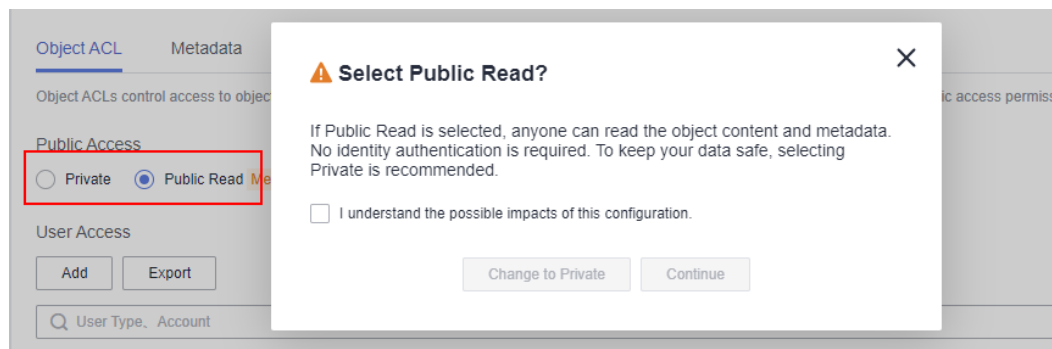
### Procedure

- Step 1** In the bucket list, click the bucket you want to operate to go to the **Objects** page.
- Step 2** Click a desired object.
- Step 3** On the **Object ACL** page, choose a permission from **Private** and **Public Read** to grant object ACL permission for anonymous users.

#### NOTE

1. After you change **Public Read** to **Private**, only the bucket owner or object owner has the access.
2. After you change **Private** to **Public Read**, anyone can read the object content and metadata. No identity authentication is required.

**Figure 2-61** Changing a public access permission



**Step 4** Click **Edit** to grant the owner, anonymous user, or other accounts required permissions for the object.

**NOTE**

ACL permissions for encrypted objects cannot be granted to registered users or anonymous users.

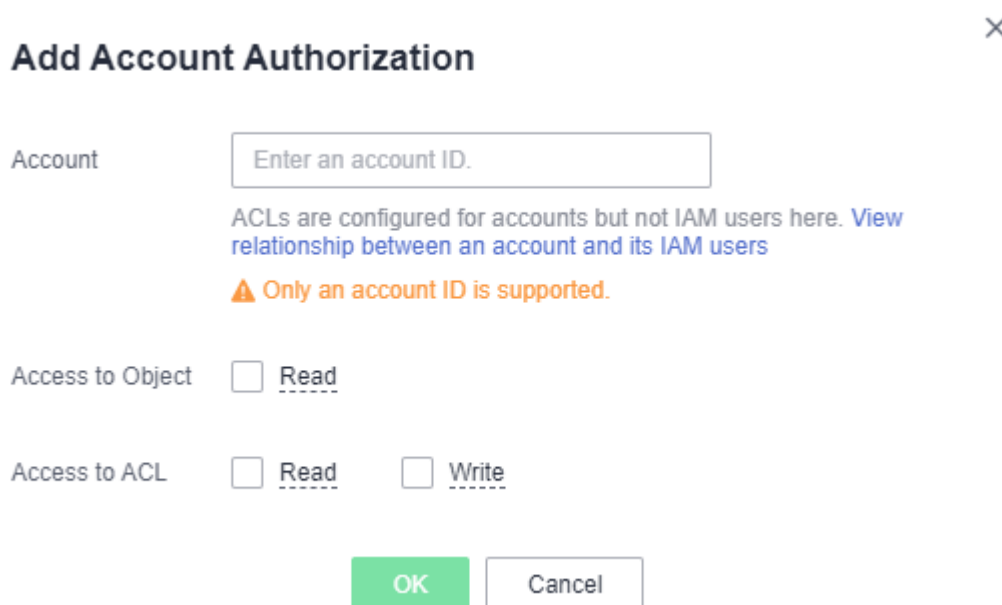
**Step 5** Click **Export** to get the object ACL configuration. The file includes the user type, account, object access, and ACL access.

**Step 6** Click **Add** to apply specific ACL permissions to an account.

Enter an account ID or account name and specify ACL permissions for the account. You can obtain the account ID or account name from the **My Credentials** page.

Click **OK**.

**Figure 2-62** Granting permissions



----End

## 2.9.9 Application Cases

### 2.9.9.1 Granting an IAM User Permissions to Operate a Specific Bucket

Create an IAM user under in an account. The IAM user has no permission to any resource before it is added to any user group. The bucket owner (root account) or other accounts and IAM users, who have the permission to set bucket policies, can configure bucket policies to grant the bucket operation permissions to IAM users.

The following is an example about how to grant an IAM user the bucket access and object upload permissions.

#### Notes

In this example, the authorized IAM user can access the authorized bucket and upload objects to the bucket using OBS Browser+, APIs, or SDKs, but cannot access the bucket on OBS Console. To allow the access through OBS Console, you need to [create a custom policy](#) to add the IAM user to the user group that has the **obs:bucket:ListAllMyBuckets** permission for all OBS resources. In this way, the IAM user can view the authorized bucket on OBS Console.

#### Procedure

- Step 1** In the bucket list, click the bucket you want to operate to go to the **Objects** page.
- Step 2** In the navigation pane, choose **Permissions > Bucket Policies**.
- Step 3** Click **Create**.
- Step 4** Configure parameters listed in the table below to grant an IAM user the permissions to access the bucket (to list objects in the bucket) and to upload objects.

**Table 2-29** Parameters for granting bucket access and object upload permissions

Parameter		Description
Configuration method		Choose <b>Visual Editor</b> .
Policy Name		Enter a custom policy name.
Policy content	Effect	Select <b>Allow</b> .
	Principals	<ul style="list-style-type: none"> <li>● Select <b>Current account</b>.</li> <li>● Specify an IAM user under the current account.</li> </ul>
	Resources	<ul style="list-style-type: none"> <li>● Method 1:                             <ul style="list-style-type: none"> <li>– Select <b>Entire bucket (including the objects in it)</b>.</li> </ul> </li> <li>● Method 2:                             <ul style="list-style-type: none"> <li>– Select <b>Current bucket</b> and <b>Specified objects</b>.</li> <li>– Set the resource path to * (indicating all objects in the bucket).</li> </ul> </li> </ul>

Parameter		Description
	Actions	<ul style="list-style-type: none"> <li>Choose <b>Customize</b>.</li> <li>Select the following actions:                             <ul style="list-style-type: none"> <li><b>ListBucket</b> (to list objects in the bucket and obtain the bucket metadata)</li> <li><b>PutObject</b> (to upload objects)</li> </ul> </li> </ul> <p><b>NOTE</b> In this example, only the upload action among object actions is selected. You can also select other object actions to grant corresponding permissions if needed. The asterisk (*) indicates all actions.</p> <p>To learn the supported actions and their meanings, see <a href="#">ActionsActions</a>.</p>

**Step 5** Click **Create** in the lower right corner.

----End

### 2.9.9.2 Granting Other Accounts Permissions to Operate a Specific Bucket

The bucket owner (root account) or other accounts and IAM users, who have the permission to set bucket policies, can configure bucket policies to grant the bucket operation permissions to other accounts or IAM users under other accounts.

The following is an example about how to grant other accounts bucket access and object upload permissions.

#### NOTE

To grant permissions to IAM users under other accounts, you need to configure both bucket policies and IAM policies.

1. Configure a bucket policy to allow IAM users to access the bucket.
2. Configure IAM policies for the account where authorized IAM users belong, to allow the IAM users to access the bucket.

Only permissions that are allowed by both the bucket policy and IAM policies can take effect.

### Procedure

**Step 1** In the bucket list, click the bucket you want to operate to go to the **Objects** page.

**Step 2** In the navigation pane, choose **Permissions > Bucket Policies**.

**Step 3** Click **Create**.

**Step 4** Configure parameters listed in the table below to grant other accounts the permissions to access the bucket (to list objects in the bucket) and to upload objects.



**Table 2-30** Parameters for granting bucket access and object upload permissions

Parameter		Description
Configuration method		Choose <b>Visual Editor</b> .
Policy Name		Enter a custom policy name.
Policy content	Effect	Select <b>Allow</b> .
	Principals	<ul style="list-style-type: none"> <li>Select <b>Other accounts</b>.</li> </ul> <p><b>NOTE</b></p> <ol style="list-style-type: none"> <li>You can obtain the account ID and IAM user ID from the <b>My Credentials</b> page.</li> <li>Accounts should be configured in the <i>Domain ID/IAM user ID</i> format, with each one on a separate line.</li> <li>The following describes different authorization scenarios: <ul style="list-style-type: none"> <li><b>Granting permissions to all the other accounts and their IAM users:</b> Set the account ID and IAM user ID to *.</li> <li><b>Granting permissions to an account:</b> Enter the desired account ID and IAM user ID.</li> <li><b>Granting permissions to an account and its IAM users:</b> Enter the desired account ID, and set the IAM user ID to * (indicating all IAM users under the account).</li> <li><b>Granting permissions to certain IAM users:</b> Enter the account ID and one or more IAM user IDs.</li> </ul> </li> </ol>
	Resources	<ul style="list-style-type: none"> <li>Method 1: <ul style="list-style-type: none"> <li>Select <b>Entire bucket (including the objects in it)</b>.</li> </ul> </li> <li>Method 2: <ul style="list-style-type: none"> <li>Select <b>Current bucket and Specified objects</b>.</li> <li>Set the resource path to * (indicating all objects in the bucket).</li> </ul> </li> </ul>
Actions	<ul style="list-style-type: none"> <li>Choose <b>Customize</b>.</li> <li>Select actions: <b>ListBucket</b> (to list objects in the bucket and obtain the bucket metadata) and <b>PutObject</b> (to upload objects).</li> </ul> <p><b>NOTE</b></p> <p>In this example, only the upload action among object actions is selected. You can also select other object actions to grant corresponding permissions if needed. The asterisk (*) indicates all actions.</p> <p>To learn the supported actions and their meanings, see <a href="#">ActionsActions</a>.</p>	

**Step 5** Click **Create** in the lower right corner.

----End

### 2.9.9.3 Restricting Access to a Bucket for Specific Addresses

You can configure a bucket policy to restrict access to a bucket for specified addresses. This example describes how to deny access from clients whose IP address is in the range of **114.115.1.0/24** to a bucket.

#### Procedure

**Step 1** In the bucket list, click the bucket you want to operate to go to the **Objects** page.

**Step 2** In the navigation pane, choose **Permissions > Bucket Policies**.

**Step 3** Click **Create**.

**Step 4** Configure parameters listed in the table below.

**Table 2-31** Restricting access to a bucket for specified addresses

Parameter		Description
Configuration method		Choose <b>Visual Editor</b> .
Policy Name		Enter a custom policy name.
Policy content	Effect	Select <b>Deny</b> .
	Principals	<ul style="list-style-type: none"><li>• Select <b>All accounts</b>.</li></ul>
	Resources	<ul style="list-style-type: none"><li>• Method 1:<ul style="list-style-type: none"><li>– Select <b>Entire bucket (including the objects in it)</b>.</li></ul></li><li>• Method 2:<ul style="list-style-type: none"><li>– Select <b>Current bucket and Specified objects</b>.</li><li>– Set the resource path to * (indicating all objects in the bucket).</li></ul></li></ul>
	Actions	<ul style="list-style-type: none"><li>• Choose <b>Customize</b>.</li><li>• Select * (indicating all actions).</li></ul>
	Conditions	<ul style="list-style-type: none"><li>• <b>Key</b>: Select <b>Sourcelp</b>.</li><li>• <b>Condition Operator</b>: Select <b>IpAddress</b>.</li><li>• <b>Value</b>: Enter <b>114.115.1.0/24</b>.</li></ul>

**Step 5** Click **Create** in the lower right corner.

----End

## Verification

Initiate an access request from an IP address in the range of **114.115.1.0/24**. The access is denied. Initiate an access request from an IP address beyond the range of **114.115.1.0/24**. The access is allowed.

### 2.9.9.4 Limiting the Time When Objects in a Bucket Are Accessible

You can configure the bucket policy to limit the time when objects in a bucket are accessible. In the following example, the access time window is from 2019-03-26T12:00:00Z to 2019-03-26T15:00:00Z.

## Procedure

- Step 1** In the bucket list, click the bucket you want to operate to go to the **Objects** page.
- Step 2** In the navigation pane, choose **Permissions > Bucket Policies**.
- Step 3** Click **Create**.
- Step 4** Configure parameters listed in the table below.

**Table 2-32** Limiting the time when objects in a bucket are accessible

Parameter		Description
Configuration method		Choose <b>Visual Editor</b> .
Policy Name		Enter a custom policy name.
Policy content	Effect	Select <b>Allow</b> .
	Principals	<ul style="list-style-type: none"> <li>● Select <b>All accounts</b>.</li> </ul>
	Resources	<ul style="list-style-type: none"> <li>● Select <b>Specified objects</b>.</li> <li>● Set the resource path to <b>*</b>.</li> </ul> <p><b>NOTE</b></p> <ol style="list-style-type: none"> <li>1. * indicates all objects in a bucket.</li> <li>2. This example only grants permissions for resources in the bucket. If you also want to grant permission for the bucket (for example, the permission to list objects in the bucket), create another custom bucket policy.</li> </ol>
	Actions	<ul style="list-style-type: none"> <li>● Choose <b>Customize</b>.</li> <li>● Select <b>*</b> (indicating all object actions).</li> </ul> <p><b>NOTE</b></p> <p>Selecting <b>*</b> may cause resources to be deleted. To avoid this risk, select <b>Get*</b> that indicates all read permissions.</p>

Parameter		Description
	Conditions	<ul style="list-style-type: none"> <li>• Condition 1:                             <ul style="list-style-type: none"> <li>- <b>Key:</b> Select <b>CurrentTime</b>.</li> <li>- <b>Condition Operator:</b> Select <b>DateGreaterThan</b>.</li> <li>- <b>Value:</b> Enter <b>2019-03-26T12:00:00Z</b> (UTC).</li> </ul> </li> <li>• Condition 2:                             <ul style="list-style-type: none"> <li>- <b>Key:</b> Select <b>CurrentTime</b>.</li> <li>- <b>Condition Operator:</b> Select <b>DateLessThan</b>.</li> <li>- <b>Value:</b> Enter <b>2019-03-26T15:00:00Z</b> (UTC).</li> </ul> </li> </ul>

**Step 5** Click **Create** in the lower right corner.

----End

## Verification

During the specified time period, any user can access the specified resources in the bucket. Outside the specified time period, only the bucket owner can access the bucket.

### 2.9.9.5 Granting Anonymous Users Permission to Access Objects

An enterprise stores a large volume of map data in OBS, and offers the data for public query. This enterprise sets a read permission for anonymous users, and provides the data URLs on the Internet. Then all users can read or download the data through the URLs.

## Procedure

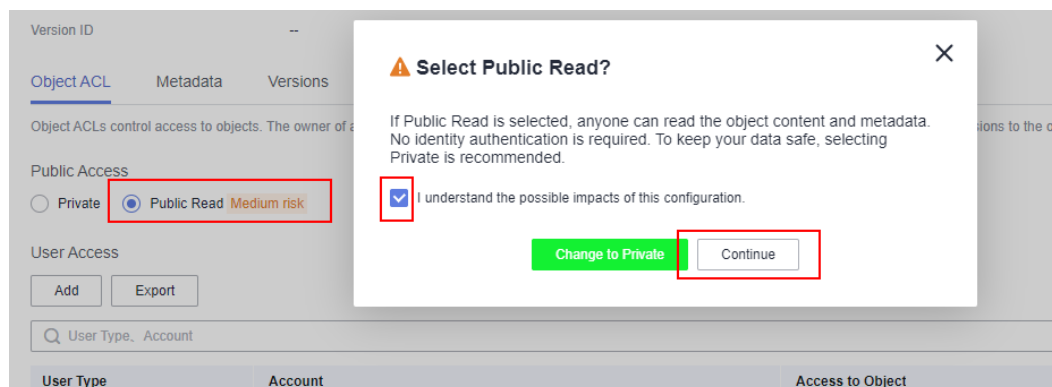
**Step 1** Log in to OBS Console and click **Create Bucket** to create a bucket.

**Step 2** In the bucket list, click the name of the newly created bucket. On the displayed object management page, upload the map data to the new bucket. The map data is stored as an object.

**Step 3** Click the object name. The object details page is displayed.

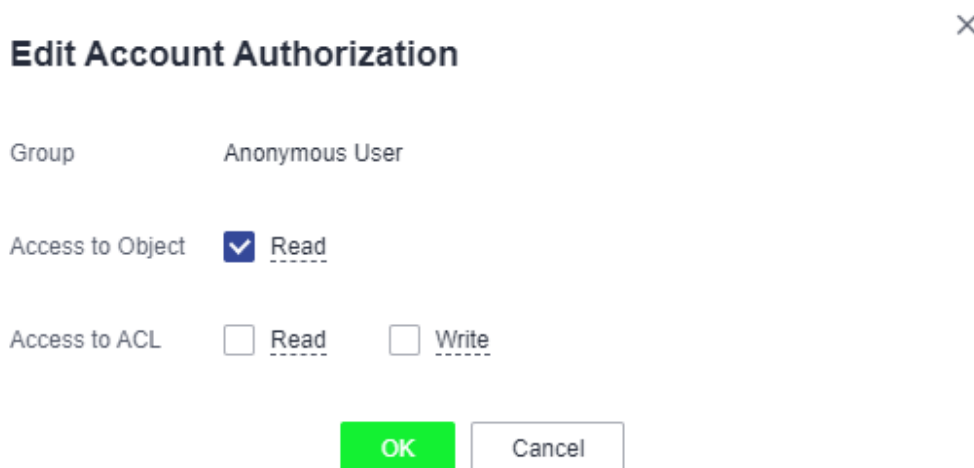
**Step 4** Choose **Object ACL > Public Access > Public Read**. In the displayed dialog box, select **I understand the possible impacts of this configuration**, and click **Continue**.

**Figure 2-63** Selecting public read



**Step 5** Click **Edit** in the **Operation** column of **Anonymous User**. In the displayed dialog box, grant the object read permission to anonymous users and click **OK**, as shown in [Figure 2-64](#).

**Figure 2-64** Granting the object read permission to anonymous users



**Step 6** Click **OK**.  
----End

## Verification

- Step 1** Click the object. Its URL is displayed under **Link**. Share the URL over the Internet, so that all users can access or download the object through the Internet.
- Step 2** An anonymous user can view the object by copying the URL of the object to the web browser.  
----End

### 2.9.9.6 Granting Anonymous Users Permission to Access Folders

If all objects in a folder need to be accessible to anonymous users, you can configure a bucket policy or an object policy to grant anonymous users the

permission to access the folder. In this example, a bucket policy is used. If you want to use an object policy to grant permission, select the target folder and configure an object policy. Parameters in both types of policies are the same.

## Procedure

- Step 1** In the bucket list, click the bucket you want to operate to go to the **Objects** page.
- Step 2** In the navigation pane, choose **Permissions > Bucket Policies**.
- Step 3** Click **Create**.
- Step 4** Configure parameters listed in the table below.

**Table 2-33** Granting folder access permissions to anonymous users

Parameter		Description
Configuration method		Choose <b>Visual Editor</b> .
Policy Name		Enter a custom policy name.
Policy content	Effect	Allow
	Principals	<ul style="list-style-type: none"> <li>● Select <b>All accounts</b>.</li> </ul>
	Resources	<ul style="list-style-type: none"> <li>● Select <b>Specified objects</b>.</li> <li>● Enter an object name prefix for the resource path, for example, <b>folder-001/*</b>, which indicates that the policy applies to all objects in folder <b>folder-001</b>.</li> </ul>
	Actions	<ul style="list-style-type: none"> <li>● Choose <b>Customize</b>.</li> <li>● Select <b>GetObject</b> (to obtain object content and metadata).</li> </ul>

- Step 5** Click **Create** in the lower right corner.

----End

## Verification

- Step 1** After the permission is successfully configured, select an object in the folder and click the object name to view its details. The object link (URL) is displayed on the details page. Share the URL over the Internet, so that all users can access or download the object through the Internet.
- Step 2** Use the URL to access the object in a browser. Anyone can access the object.

----End

## 2.10 Versioning

## 2.10.1 Versioning Overview

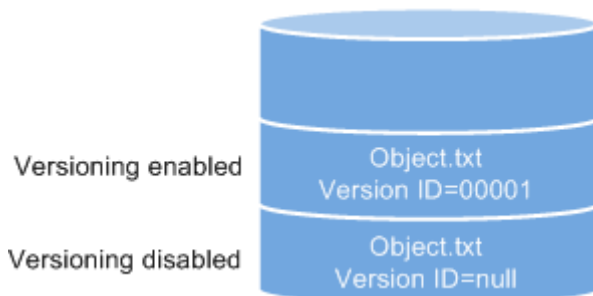
OBS can store multiple versions of an object. You can quickly search for and restore different versions or restore data in the event of accidental deletions or application faults.

By default, the versioning function is disabled for new buckets on OBS. Therefore, if you upload an object to a bucket where an object with the same name exists, the new object will overwrite the existing one.

### Enabling Versioning

- Enabling versioning does not change the versions and contents of existing objects in the bucket. The version ID of an object is **null** before versioning is enabled. If a namesake object is uploaded after versioning is enabled, a version ID will be assigned to the object. For details, see [Figure 2-65](#).

**Figure 2-65** Versioning (with existing objects)



- OBS automatically allocates a unique version ID to a newly uploaded object. Objects with the same name are stored in OBS with different version IDs.

**Figure 2-66** Versioning (for new objects)



**Table 2-34** Version description

Version	Description
Latest version	After versioning is enabled, each operation on an object will result in saving of the object with a new version ID. The version ID generated upon the latest operation is called the latest version.

Version	Description
Historical version	After versioning is enabled, each operation on an object will result in saving of the object with a new version ID. Version IDs generated upon operations other than the latest operation are called historical versions.

- The latest objects in a bucket are returned by default after a GET Object request.
- Objects can be downloaded by version IDs. By default, the latest object is downloaded if the version ID is not specified. For details, see [Related Operations](#) in [Configuring Versioning](#).
- You can select an object and click **Delete** on the right to delete the object. After the object is deleted, OBS generates a **Delete Marker** with a unique version ID for the deleted object, and the deleted object is displayed in the **Deleted Objects** list. For details, see [Deleting an Object or Folder](#). The 404 error will be returned if attempts are made to access this deleted object.

**Figure 2-67** Object with a delete marker

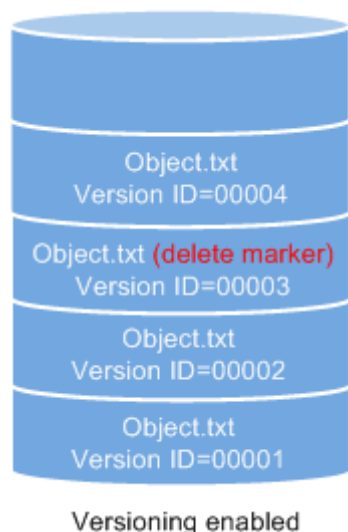


- You can recover a deleted object by deleting the delete marker. For details, see [Related Operations](#) in [Undeleting an Object](#).
- After an object is deleted, you can specify the version number in **Deleted Objects** to permanently delete the object of the specified version. For details, see [Related Operations](#) in [Deleting an Object or Folder](#).
- An object is displayed either in the object list or the list of deleted objects. It will never be displayed in both the lists at the same time.

For example, after object **A** is uploaded and deleted, it will be displayed in the **Deleted Objects** list. If you upload an object named **A** again, the object **A** will be displayed in the **Objects** list, and the previously deleted object **A** will no longer be displayed in the **Deleted Objects** list. For details, see [Figure 2-68](#).



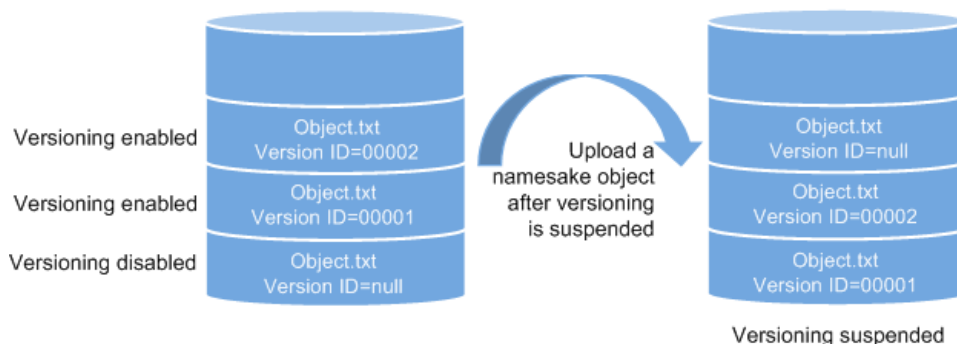
**Figure 2-68** Uploading a namesake object after the original one is deleted



### Suspending Versioning

Once the versioning function is enabled, it can be suspended but cannot be disabled. Once versioning is suspended, version IDs will no longer be allocated to newly uploaded objects. If an object with the same name already exists and does not have a version ID, the object will be overwritten.

**Figure 2-69** Object versions in the scenario when versioning is suspended



If versions of objects in a bucket do not need to be controlled, you can suspend the versioning function.

- Historical versions will be retained in OBS. If you do not need these historical versions, manually delete them.
- Objects can be downloaded by version IDs. By default, the latest object is downloaded if the version ID is not specified.

### Differences Between Scenarios When Versioning Is Suspended and Disabled

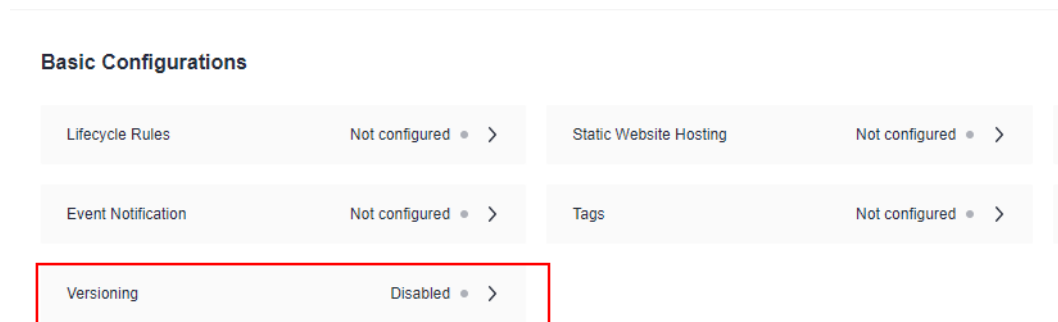
If you delete an object after versioning is suspended for the bucket, a delete marker will be generated, no matter whether the object has historical versions. But, if versioning is disabled, the same operation will not generate a delete marker.

## 2.10.2 Configuring Versioning

### Procedure

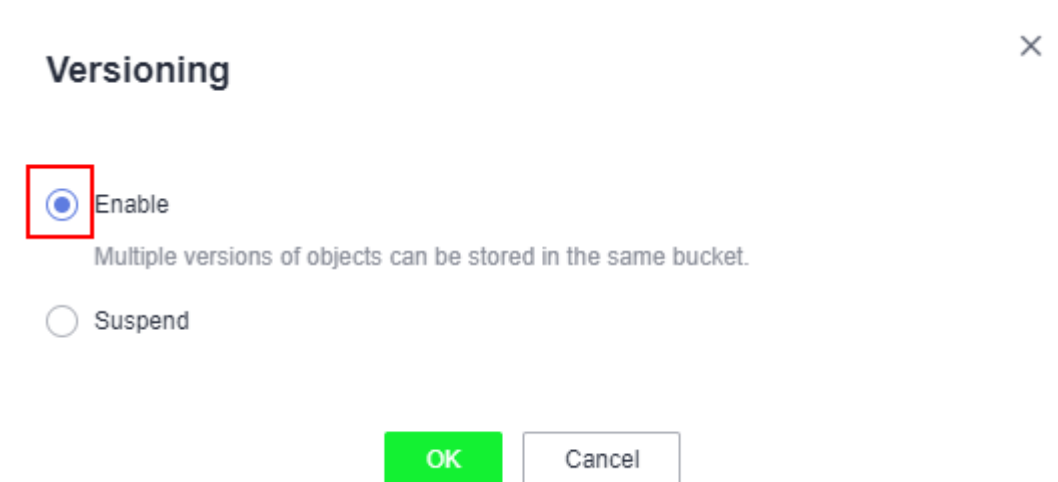
- Step 1** In the bucket list, click the bucket you want to operate to go to the **Objects** page.
- Step 2** In the navigation pane, choose **Overview**.
- Step 3** In the **Basic Configurations** area, click **Versioning**.

**Figure 2-70** Editing versioning status

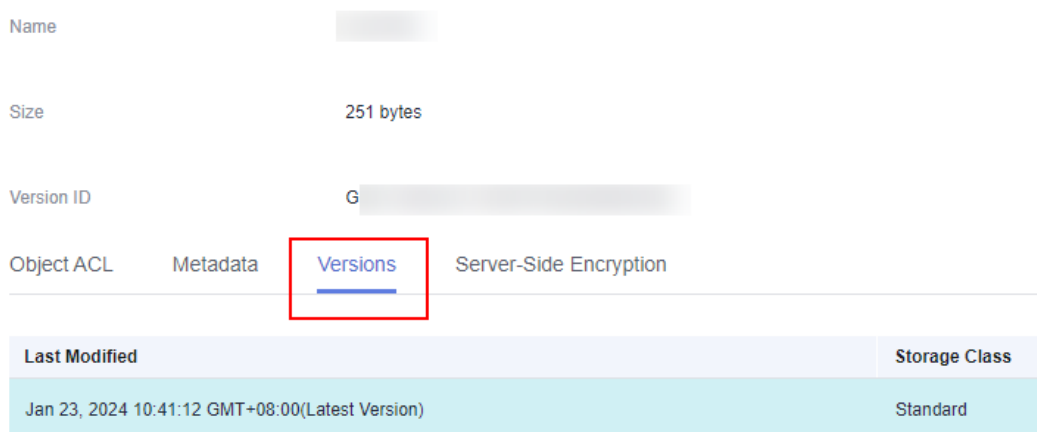


- Step 4** Select **Enable**.

**Figure 2-71** Configuring versioning



- Step 5** Click **OK** to enable versioning for the bucket.
- Step 6** Click an object to go to the object details page. On the **Versions** tab, view all versions of the object.

**Figure 2-72** Viewing object versions

The screenshot shows the details of an object in the OBS console. The 'Versions' tab is highlighted with a red box. Below the tabs is a table with two columns: 'Last Modified' and 'Storage Class'. The table contains one row with the following data:

Last Modified	Storage Class
Jan 23, 2024 10:41:12 GMT+08:00(Latest Version)	Standard

----End

## Related Operations

After versioning is configured for a bucket, you can go to the object details page, click the **Versions** tab, and then delete and download object versions.

**Step 1** In the bucket list, click the bucket you want to operate to go to the **Objects** page.

**Step 2** In the object list, click the object you want to go to the object details page.

**Step 3** On the **Versions** tab, view all versions of the object.

**Step 4** Perform the following operations on object versions:

1. Download a desired version of the object by clicking **Download** in the **Operation** column.

### NOTE

If the version you want to download is in the Cold storage class, restore it first.

2. Delete a version of the object by clicking **Delete** in the **Operation** column. If you delete the latest version, the most recent version becomes the latest version.

----End

## 2.11 Logging

### 2.11.1 Logging Overview

You can enable logging to facilitate analysis or audit as required. Access logs enable a bucket owner to analyze the property, type, or trend of requests to the bucket in depth. When the logging function of a bucket is enabled, OBS will log access requests for the bucket automatically, and write the generated log files to the specified bucket (target bucket).

After logging is enabled, the log delivery user group will be automatically granted the permission to read the bucket ACL and write the bucket where logs are saved. If you manually disable such permissions, bucket logging fails.

OBS can record bucket access requests in logs for request analysis and log audit.

Logs occupy the OBS storage that incurs costs, so OBS does not collect bucket access logs by default.

OBS creates log files and uploads them to a specified bucket. To perform these operations, OBS must be granted required permissions. Therefore, before configuring logging for a bucket, you need to create an IAM agency for OBS and add this agency when configuring logging for the bucket. By default, when configuring permissions for an agency, you only need to grant the agency the permission to upload log files (PutObject) to the bucket for storing log files. In the following example, **mybucketlogs** is the bucket. If the default encryption function is enabled for the log storing bucket, the IAM agency also requires the KMS Administrator permissions in the region where the log storing bucket resides.

```
{
  "Version": "1.1",
  "Statement": [
    {
      "Action": [
        "obs:object:PutObject"
      ],
      "Resource": [
        "OBS:*:*:object:mybucketlogs/*"
      ],
      "Effect": "Allow"
    }
  ]
}
```

After logging is configured, you can view operation logs in the bucket that stores the logs in approximately fifteen minutes.

The following shows an example access log of the target bucket:

```
787f2f92b20943998a4fe2ab75eb09b8 bucket [13/Aug/2015:01:43:42 +0000] xx.xx.xx.xx
787f2f92b20943998a4fe2ab75eb09b8 281599BACAD9376ECE141B842B94535B
REST.GET.BUCKET.LOCATION
- "GET /bucket?location HTTP/1.1" 200 - 211 - 6 6 "-" "HttpClient" - -
```

The access log of each bucket contains the following information.

**Table 2-35** Bucket log format

Parameter	Value Example	Description
BucketOwner	787f2f92b20943998a4fe2ab75eb09b8	Account ID of the bucket owner
Bucket	bucket	Name of the bucket
Time	[13/Aug/2015:01:43:42 +0000]	Timestamp of the request (UTC)

Parameter	Value Example	Description
Remote IP	xx.xx.xx.xx	IP address from where the request is initiated
Requester	787f2f92b20943998a4fe2ab75eb09b8	Requester ID
RequestID	281599BACAD9376ECE141B842B94535B	Request ID
Operation	REST.GET.BUCKET.LOCATION	Name of the operation
Key	-	Object name
Request-URI	GET /bucket?location HTTP/1.1	URI of the request
HTTPStatus	200	Return code
ErrorCode	-	Error code
BytesSent	211	Size of the HTTP response, expressed in bytes
ObjectSize	-	Object size (bytes)
TotalTime	6	Processing time on the server (ms)
Turn-AroundTime	6	Total time for processing the request (ms)
Referer	-	Header field <b>Referer</b> of the request
User-Agent	HttpClient	User-Agent header of the request
VersionID	-	Version ID carried in the request
STSLogUrn	-	Federated authentication and agency information
StorageClass	STANDARD_IA	Current storage class of the object
TargetStorageClass	GLACIER	Storage class that the object will be transited to

## 2.11.2 Configuring Access Logging for a Bucket

After logging is enabled for a bucket, OBS automatically converts bucket logs into objects following the naming rules and writes the objects into a target bucket.

### Procedure

- Step 1** In the bucket list, click the bucket you want to operate to go to the **Objects** page.
- Step 2** In the navigation pane, choose **Overview**.
- Step 3** In the **Basic Configurations** area, click **Logging**. The **Logging** dialog box is displayed.
- Step 4** Select **Enable**.

Figure 2-73 Logging

**Logging** ×

i Access requests can be logged for analysis or auditing. [Learn more](#)

**Enable**

The log delivery user will be automatically granted permissions to read the ACL of the bucket where logs are to be saved and write logs to the bucket. Uploading logs to buckets will generate billable PUT requests. For details, see OBS Pricing Details.

Save Logs To  ↻ ?

Log File Name Prefix  ?

IAM Agency  ↻ [Create Agency](#) ?

**Disable**

- Step 5** Select an existing bucket where you want to store log files. Log delivery users of the selected bucket will be automatically granted the permissions to read the bucket ACL and write logs to the bucket.

- Step 6** Enter a prefix for the **Log File Name Prefix**.

After logging is enabled, generated logs are named in the following format:

*<Log File Name Prefix>YYYY-mm-DD-HH-MM-SS-<UniqueString>*

- *<Log File Name Prefix>* is the shared prefix of log file names.
- **YYYY-mm-DD-HH-MM-SS** indicates when the log is generated.
- *<UniqueString>* indicates a character string generated by OBS.

On OBS Console, if the configured *<Log File Name Prefix>* ends with a slash (/), logs generated in the bucket are stored in the *<Log File Name Prefix>* folder in the bucket, facilitating the management of log files.

Example:

- If the bucket named **bucket** is used to save log files, and the log file name prefix is set to **bucket-log/**, all log files delivered to this bucket are saved in the **bucket-log** folder. A log file is named as follows: **2015-06-29-12-22-07-N7MXLAF1BDG7MPDV**.
- If the bucket named **bucket** is used to save log files, and the log file name prefix is set to **bucket-log**, all log files are saved in the root directory of the bucket. A log file is named as follows: **bucket-log2015-06-29-12-22-07-N7MXLAF1BDG7MPDV**.

**Step 7** Select an IAM agency to grant OBS the permission to upload log files to the specified bucket.

By default, when configuring permissions for an agency, you only need to grant the agency the permission to upload log files (PutObject) to the bucket for storing log files. In the following example, **mybucketlogs** is the bucket. If default encryption is enabled for the log storage bucket, the IAM agency also requires the **KMS Administrator** permission for the region where the bucket is located.

```
{
  "Version": "1.1",
  "Statement": [
    {
      "Action": [
        "obs:object:PutObject"
      ],
      "Resource": [
        "OBS:*:*:object:mybucketlogs/*"
      ],
      "Effect": "Allow"
    }
  ]
}
```

You can choose an existing IAM agency from the drop-down list or click **Create Agency** to create one. For details about how to create an agency, see [Creating an IAM Agency](#).

**Step 8** Click **OK**.

After logging is configured, you can view operation logs in the bucket that stores the logs in approximately fifteen minutes.

----End

## Related Operations

If you do not need to record logs, click **Disable** in the **Logging** dialog box and then click **OK**. After logging is disabled, logs are not recorded, but existing logs in the target bucket will be retained.

## 2.12 Tags

### 2.12.1 Tag Overview

Tags are used to identify and classify OBS buckets.

If you add tags to a bucket, service detail records (SDRs) generated for it will be labeled with these tags. You can classify SDRs by tag for cost analysis. For example, if you have an application that uploads its running data to a bucket, you can tag the bucket with the application name. In this manner, the costs on the application can be analyzed using tags in SDRs.

A tag is described using a key-value pair. A bucket can have a maximum of 10 tags. Each tag has only one key and one value.

The key and value can exist in either sequence in a tag. Each key is unique among all tags of a bucket, whereas values can be repetitive or blank.

### 2.12.2 Configuring Tags for a Bucket

When creating a bucket, you can add tags to it. For details, see [Creating a Bucket](#). You can also add tags to a bucket after it has been created. This topic describes how to add tags to an existing bucket.

#### Procedure

- Step 1** In the bucket list, click the bucket you want to operate to go to the **Objects** page.
- Step 2** In the navigation pane, choose **Overview**.
- Step 3** In the **Basic Configurations** area, click **Tags**.  
Alternatively, you can choose **Basic Configurations** > **Tagging** in the navigation pane.
- Step 4** Click **Add Tag**. The **Add Tag** dialog box is displayed.

Figure 2-74 Add Tag



**Step 5** Set the key and value based on [Table 2-36](#).

**Table 2-36** Parameter description

Parameter	Description
Tag key	Key of a tag. Tag keys for the same bucket must be unique. You can customize tags or select the ones predefined on TMS. A tag key: <ul style="list-style-type: none"><li>• Must contain 1 to 36 characters and be case sensitive.</li><li>• Can contain only digits, letters, underscores (_), and hyphens (-).</li></ul>
Tag value	Value of a tag. A tag value can be repetitive or left blank. A tag value: <ul style="list-style-type: none"><li>• Can contain 0 to 43 characters and must be case sensitive.</li><li>• Can contain only digits, letters, underscores (_), and hyphens (-).</li></ul>

**Step 6** Click **OK**.

It takes approximately 3 minutes for the tag to take effect.

----**End**

## Related Operations

In the tag list, click **Edit** to change the tag value or click **Delete** to remove the tag.

## 2.13 Event Notifications

### 2.13.1 SMN-Enabled Event Notifications

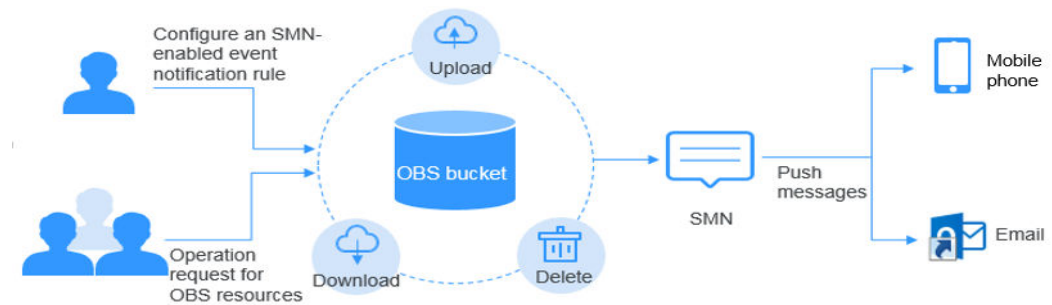
Simple Message Notification (SMN) is a reliable and extensible message notification service that can handle a huge number of messages. It significantly simplifies system coupling and can automatically push messages to endpoints via email.

OBS leverages SMN to provide event notifications. In OBS, you can use SMN to send event notifications to specified subscribers, so that you will be informed of any critical operations (such as upload and deletion) that occur on specified buckets in real time. For example, you can configure an event notification rule to send messages through SMN to the specified email address whenever an upload operation occurs on the specified bucket.

You can configure the event notification rule to filter objects by the object name prefix or suffix. For example, you can add an event notification rule to send notifications whenever an object with the **.jpg** suffix is uploaded to the specified bucket. You can also add an event notification rule to send notifications whenever an object with the **images/** prefix is uploaded to the specified bucket.

For details about events supported by SMN and how to configure an SMN-enabled event notification rule, see [Configuring SMN-Enabled Event Notification](#).

Figure 2-75 SMN-enabled event notification



## 2.13.2 Configuring SMN-Enabled Event Notification

This topic describes how to configure an SMN-enabled event notification rule on OBS Console.

### Background Information

For details, see [SMN-Enabled Event Notifications](#).

### Procedure

- Step 1** In the bucket list, click the bucket you want to operate to go to the **Objects** page.
- Step 2** In the navigation pane, choose **Overview**.
- Step 3** In the **Basic Configurations** area, click **Event Notification**. The **Event Notification** page is displayed.  
  
Alternatively, you can choose **Basic Configurations** > **Event Notification** in the navigation pane.
- Step 4** Click **Create**. The **Create Event Notification** dialog box is displayed.

**Figure 2-76** Creating an event notification rule

**Create Event Notification**
×

Name  ?

Events  ?

Prefix  ?

Suffix  ?

Notification Method  SMN topic ?

↻

↻ Create Topic

**Step 5** Configure event notification parameters, as described in [Table 2-37](#).

**Table 2-37** Event notification parameters

Parameter	Description
Name	Name of the event. If the event name is left blank, the system will automatically assign a globally unique ID.

Parameter	Description
Events	<p>Various types of events. Currently, OBS supports event notification for the following types of events:</p> <ul style="list-style-type: none"> <li>● <b>ObjectCreated</b>: Indicates all kinds of object creation operations, including PUT, POST, and COPY of objects, as well as the merging of parts. <ul style="list-style-type: none"> <li>– <b>Put</b>: Creates or overwrites an object using the PUT method.</li> <li>– <b>Post</b>: Creates or overwrites an object using the POST (browser-based upload) method.</li> <li>– <b>Copy</b>: Creates or overwrites an object using the COPY method.</li> <li>– <b>CompleteMultipartUpload</b>: Merges parts of a multipart upload.</li> </ul> </li> <li>● <b>ObjectRemoved</b>: Deletes an object. <ul style="list-style-type: none"> <li>– <b>Delete</b>: Deletes an object with a specified version ID.</li> <li>– <b>DeleteMarkerCreated</b>: Deletes an object without specifying a version ID.</li> </ul> </li> </ul> <p>Multiple event types can be applied to the same object. For example, if you have selected <b>Put</b>, <b>Copy</b>, and <b>Delete</b> in the same event notification rule, a notification will be sent to you when the specified object is uploaded to, copied to, or deleted from the bucket. <b>ObjectCreated</b> contains <b>Put</b>, <b>Post</b>, <b>Copy</b>, and <b>CompleteMultipartUpload</b>. If you select <b>ObjectCreated</b>, the events <b>ObjectCreated</b> contains are automatically selected. Similarly, if you select <b>ObjectRemoved</b>, <b>Delete</b> and <b>DeleteMarkerCreated</b> are automatically selected.</p>
Prefix	<p>Object name prefix for which notifications will be triggered.</p> <p><b>NOTE</b> If neither the <b>Prefix</b> nor the <b>Suffix</b> is configured, the event notification rule applies to all objects in the bucket.</p>
Suffix	<p>Object name suffix for which notifications will be triggered.</p> <p><b>NOTE</b></p> <ul style="list-style-type: none"> <li>● A folder path ends with a slash (/). Therefore, if you want to configure the event notification for operations on folders and you need to filter folders by suffix, the suffix must also end with a slash (/).</li> <li>● If neither the <b>Prefix</b> nor the <b>Suffix</b> is configured, the event notification rule applies to all objects in the bucket.</li> </ul>

Parameter	Description
SMN Topic	Project: The project that contains the SMN topic you want to select.  Projects are used to manage and classify cloud resources, including SMN topics. Each project contains different SMN topics. Select a project first and then a topic.
	Topic: specifies the SMN topic that authorizes OBS to publish messages. You can create such topics on the SMN management console. <b>NOTE</b> <ul style="list-style-type: none"><li>Once SMN topics are selected for pushing OBS event notifications, do not delete them or cancel their authorizations to OBS.</li><li>If the topics are deleted or their authorizations to OBS are canceled, the following conditions may occur:<ol style="list-style-type: none"><li>The subscriber of the topic cannot receive messages.</li><li>Event notifications associated with unavailable topics are automatically cleared.</li></ol></li><li>For details about how to use SMN, see sections "Creating a Topic", "Adding a Subscription", and "Configuring Topic Policies" in the <i>Simple Message Notification User Guide</i>.</li></ul>

**Step 6** Click **OK**.

----End

## Related Operations

You can click **Edit** in the **Operation** column of an event notification rule, to edit the notification rule, or click **Delete** to delete the rule.

If you want to batch delete event notification rules, select them and click **Delete** above the list.

## 2.13.3 Application Example: Configuring SMN-Enabled Event Notification

### Background Information

An enterprise has a large number of files to archive but it does not want to cost much on storage resources. Therefore, the enterprise subscribes to OBS for storing daily files and expects that an employee can be informed of every operation performed on OBS via email.

### Procedure


**Step 1** Log in to OBS Console as an enterprise user.

**Step 2** Create a bucket.

Click **Create Bucket** in the upper right corner of the page. On the page, select a region and storage class, and specify a bucket name and other parameters. Then, click **Create Now**.

**Step 3** Create a folder.

Click the name of the bucket created in [Step 2](#) to go to the **Objects** page. Then, click **Create Folder**. In the displayed dialog box, enter a folder name and click **OK**. In the following example, **SMN** is the folder name.

**Step 4** In the upper left corner of the page, click  and choose **Simple Message Notification**. On the displayed SMN page, create a topic.

In the following example, **TestTopic** is the SMN topic and the notifications are sent via email.

Use SMN to create a notification topic for OBS as follows:

1. Create an SMN topic.
2. Add a subscription.
3. Modify the topic policy. On the **Configure Topic Policy** page, select **OBS** under **Services that can publish messages to this topic**.

For details, see [Table 2-37](#).

**Step 5** Go back to OBS Console.**Step 6** Configure an event notification rule.

1. In the bucket list, click the bucket that you have created in [Step 2](#).
2. In the navigation pane, choose **Basic Configurations > Event Notification**. The **Event Notification** page is displayed.
3. Click **Create**. The **Create Event Notification** dialog box is displayed.
4. Configure event notification parameters.

After the notification is configured, an employee will be informed of all specified operations on the **SMN** folder in bucket **testbucket**.

**Table 2-38** Event notification parameters

Parameter	Value
Name	test
Events	ObjectCreated, ObjectRemoved

Parameter	Value
Prefix	SMN/ <b>NOTE</b> <ul style="list-style-type: none"><li>- A folder path ends with a slash (/). Therefore, if you want to configure the event notification for operations on folders and you need to filter folders by suffix, the suffix must also end with a slash (/).</li><li>- If neither the <b>Prefix</b> nor the <b>Suffix</b> is configured, the event notification rule applies to all objects in the bucket.</li></ul>
Notification Method	SMN topic <i>Select the project to which the SMN topic belongs.</i> TestTopic

----End

## Verification

**Step 1** Log in to OBS Console as an enterprise user.

**Step 2** Upload the **test.txt** file to the folder created in [Step 3](#).

After the file is uploaded, an employee receives an email. Keyword **ObjectCreated:Post** in the email indicates that the object is successfully uploaded.

**Step 3** Delete the **test.txt** file uploaded in [Step 2](#).

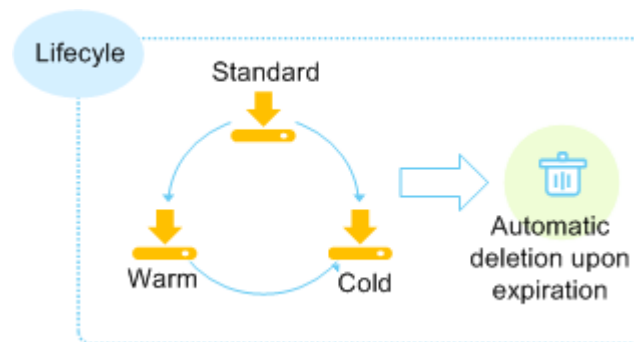
After the file is successfully deleted, an employee will receive an email. Keyword **ObjectRemoved>Delete** in the email indicates that the object is successfully deleted.

----End

## 2.14 Lifecycle Management

### 2.14.1 Lifecycle Management Overview

Lifecycle management means periodically deleting objects in a bucket or transitioning between object storage classes by configuring rules.

**Figure 2-77** Lifecycle management

You may configure lifecycle rules to:

- Periodically delete logs that are only meant to be retained for a specific period of time (a week or a month).
- Transition documents that are seldom accessed to the Warm or Cold storage class or delete them.

You can define lifecycle rules for your scenarios similar to those mentioned above to better manage your objects.

You can configure lifecycle rules for objects that will no longer be frequently accessed to transition them to the Warm or Cold storage class as needed. This can help reduce costs on storage. In short, transition basically means that the object storage class is altered without copying the object. You can also manually change the storage class of an object on the Objects page. For details, see [Uploading an Object](#).

Lifecycle rules have the following key elements:

- Policy  
You can specify an object name prefix to apply a lifecycle rule to a set of objects. You can also apply a lifecycle rule to the entire bucket (including the objects in it).
- Time  
You can specify the number of days after which objects that have been last updated and meet specified conditions are automatically transitioned to Warm or Cold, or are expired and then deleted.
  - Transition to Warm: This defines the number of days since the last object update after which objects meeting specified conditions are automatically transitioned to the Warm storage class.
  - Transition to Cold: This defines the number of days since the last object update after which objects meeting specified conditions are automatically transitioned to the Cold storage class.
  - Expiration time: This defines the number of days since the last object update after which objects meeting specified conditions are automatically expired and then deleted.

Objects can be transitioned to Warm at least 30 days after their last update. If you configure to transition objects first to Warm and then Cold, the objects must stay Warm at least 30 days before they can be transitioned to Cold. For example, if you configure to transition objects to Warm 33 days after their last update, the objects



can be transitioned to Cold at least 63 days after their last update. If only transition to Cold is used, but transition to Warm is not, there is no limit on the number of days for transition. The number set for expiration time must be larger than that specified for any of the transition operations.

## 2.14.2 Configuring a Lifecycle Rule

You can configure a lifecycle rule for a bucket or a set of objects to:

- Transition objects from Standard to Warm or Cold.
- Transition objects from Warm to Cold.
- Expire objects and then delete them.

Lifecycle rules do not transition Cold objects to other storage classes.

### Procedure

**Step 1** In the bucket list, click the bucket you want to operate to go to the **Objects** page.

**Step 2** In the navigation pane, choose **Overview**.

**Step 3** In the **Basic Configurations** area, click **Lifecycle Rules**. The **Lifecycle Rules** page is displayed.

Alternatively, you can choose **Basic Configurations > Lifecycle Rules** in the navigation pane.

**Step 4** Click **Create**.

**Figure 2-78** Creating a lifecycle rule

**Create Lifecycle Rule** [Learn more](#) ×

**i** The minimum billing units for Infrequent Access and Archive storage are, respectively, 30 or 90 days. If an Infrequent Access or Archive object is transitioned to another storage class or removed before this length of time has elapsed, you will still be billed for the minimum 30 or 90 days. ×

Once a lifecycle rule is enabled, objects under the rule will be transitioned to the specified storage class or deleted automatically after the specified expiration time. As a result, your costs may change due to changes of storage space and storage classes.

**Basic Information**

Status  Enable  Disable

Rule Name

Prefix  ?

---

**Current Version**

Transition to Infrequent Access After   ?  
(Days)

?

### Step 5 Configure a lifecycle rule.

#### Basic Information:

- **Status:**  
Select **Enable** to enable the lifecycle rule.
- **Rule Name:**  
It identifies a lifecycle rule. A rule name can contain a maximum of 255 characters.
- **Prefix:** It is optional.
  - If this field is configured, objects with the specified prefix will be managed by the lifecycle rule. The prefix cannot start with a slash (/) or contain two consecutive slashes (//), and cannot contain the following special characters: \:\*?"<>|
  - If this field is not configured, all objects in the bucket will be managed by the lifecycle rule.

#### NOTE

- If the specified prefix is overlapping with the prefix set in an existing lifecycle rule, OBS regards these two rules as one and forbids you to configure the one you are configuring. For example, if there is already a rule with prefix **abc** in OBS, you cannot configure another rule whose prefix starts with **abc**.
- If there is already a lifecycle rule based on an object prefix, you are not allowed to configure another rule that is applied to the entire bucket.
- If a lifecycle rule has been configured for the entire bucket, no more rules that apply to object name prefix can be added.

#### Current Version or Historical Version:

#### NOTE

- **Current Version** and **Historical Version** are two concepts for versioning. If versioning is enabled for a bucket, uploading objects with the same name to the bucket creates different object versions. The last uploaded object is called the current version, while those previously uploaded are called historical versions.
- You can configure either the **Current Version** or **Historical Version**, or both of them.
- **Transition to Warm:** After this number of days since the last update, objects meeting specified conditions will be transitioned to Warm. This number must be at least 30.
- **Transition to Cold:** After this number of days since the last update, objects meeting specified conditions will be transitioned to Cold. If you configure to transition objects first to Warm and then Cold, the objects must stay Warm at least 30 days before they can be transitioned to Cold. If only transition to Cold is used, but transition to Warm is not, there is no limit on the number of days for transition.
- **Delete Objects After (Days):** After this number of days since the last update, objects meeting certain conditions will be expired and then deleted. This number must be larger than that specified for any of the transition operations.

For example, on January 7, 2015, you saved the following files in OBS:

- log/test1.log

- log/test2.log
- doc/example.doc
- doc/good.txt

On January 10, 2015, you saved another four files:

- log/clientlog.log
- log/serverlog.log
- doc/work.doc
- doc/travel.txt

On January 10, 2015, you set the objects prefixed with **log** to expire one day later. You might encounter the following situations:

- Objects **log/test1.log** and **log/test2.log** uploaded on January 7, 2015 might be deleted after the last system scan. The deletion could happen on January 10, 2015 or January 11, 2015, depending on the time of the last system scan.
- Objects **log/clientlog.log** and **log/serverlog.log** uploaded on January 10, 2015 might be deleted on January 11, 2015 or January 12, 2015, depending on whether they have been stored for over one day (since their last update) when the system scan happened.

On the day of operation, you can set the objects with the name prefix **log** to be transitioned to **Warm** 30 days later, transitioned to **Cold** 60 days later, and deleted 100 days later, then OBS will transition **log/clientlog.log**, **log/serverlog.log**, **log/test1.log**, and **log/test2.log** to **Warm** when their storage duration exceeds 30 days, transition them to **Cold** when their storage duration exceeds 60 days, and delete them when their storage duration exceeds 100 days, respectively.

 **NOTE**

In theory, it takes 24 hours at most to execute a lifecycle rule. Because OBS calculates the lifecycle of an object from the next 00:00 (UTC time) after the object is uploaded, there may be a delay in transitioning objects between storage classes and deleting expired objects. Generally, the delay does not exceed 48 hours. If you make changes to an existing lifecycle rule, the rule will take effect again.

**Step 6** Click **OK** to complete the lifecycle rule configuration.

----End

## Follow-up Procedure

You can click **Edit** in the **Operation** column of a lifecycle rule to edit the rule. You can also click **Disable** or **Enable** to disable or enable it.

If you want to delete more than one lifecycle rule at a time, select them and click **Delete** above the list.

## 2.15 Configuring User-Defined Domain Names

## 2.15.1 Overview

### Application Scenario

After you upload a file to a bucket, you can access this file using the bucket's access domain name by default. If you want to use a custom domain name to access the file, bind the custom domain name to the bucket.

Assume that you have a domain name **www.example.com** and you upload an image **image.png** to an OBS bucket. As long as you bind **www.example.com** to the bucket, you can use **http://www.example.com/image.png** to access **image.png**. The steps below describe the configurations:

1. Create a bucket on OBS and upload file **image.png** to the bucket.
2. On OBS Console, bind **www.example.com** to the created bucket.
3. On the DNS server, add a CNAME record and map **www.example.com** to the domain name of the bucket.
4. Send a request for image **image.png**. After the request for **http://www.example.com/image.png** reaches OBS, OBS finds the mapping between the **www.example.com** and the bucket's domain name, and redirects the request to the **image.png** file stored in the bucket. This way, a request for **http://www.example.com/image.png** actually accesses **http://Bucket domain name/image.png**.

### Limitations and Constraints

1. Only buckets with version 3.0 or later support user-defined domain name configuration. The version number of a bucket is displayed in the **Basic Information** area.
2. By default, a bucket can have up to 20 user-defined domain names bound.
3. User-defined domain names currently allow requests over only HTTP, but not HTTPS.
4. A user-defined domain name can be bound to only one bucket.
5. The suffix of a user-defined domain name can contain 2 to 6 uppercase or lowercase letters.

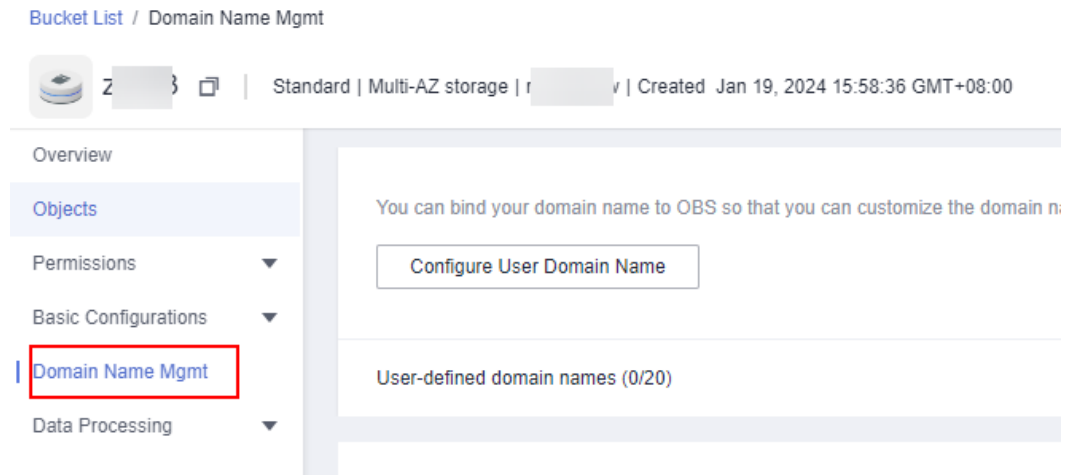
## 2.15.2 Configuring a User-Defined Domain Name

### Procedure

**Step 1** In the bucket list, click the bucket you want to operate to go to the **Objects** page.

**Step 2** In the navigation pane, choose **Domain Name Mgmt**.

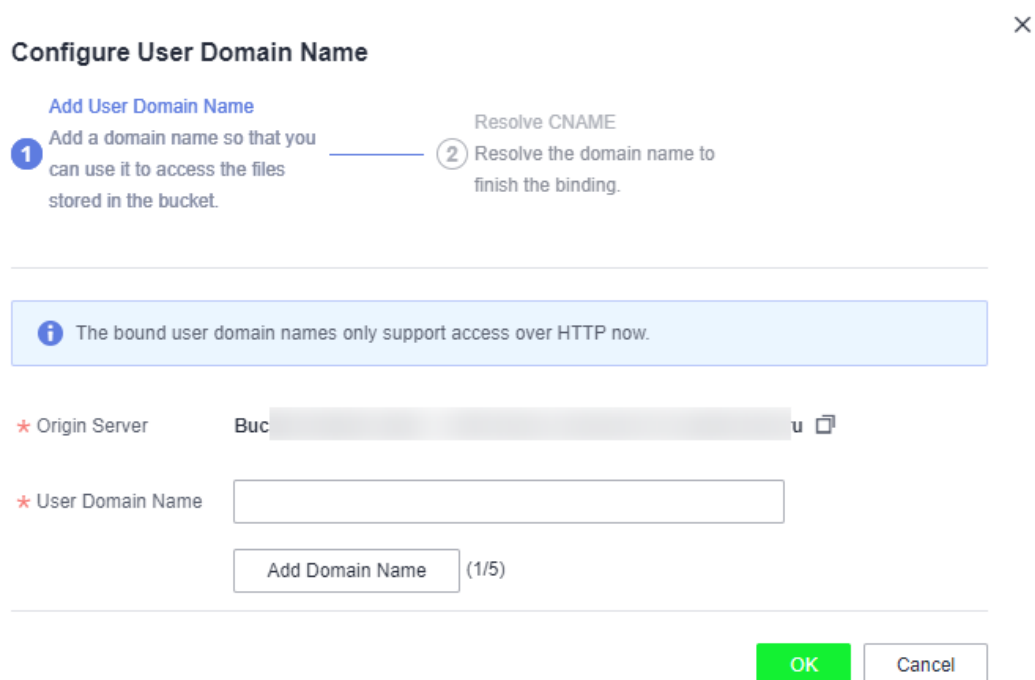
**Figure 2-79** Domain name management page



**Step 3** Click **Configure User Domain Name** in the upper part of the page. Alternatively, click **Configure User Domain Name** in the lower card area of the page when no user-defined domain names are available. In the displayed dialog box, enter the domain name to configure, as shown in **Figure 2-80**.

The suffix of a user-defined domain name can contain 2 to 6 uppercase or lowercase letters.

**Figure 2-80** Configuring a user domain name



**Step 4** Click **OK**.

**Step 5** Configure a CNAME record on the DNS, and map the user-defined domain name (for example, **example.com**) to the domain name of the bucket.

The CNAME configuration varies depending on DNS providers. For details, contact your DNS provider.

----End

## 2.16 Static Website Hosting

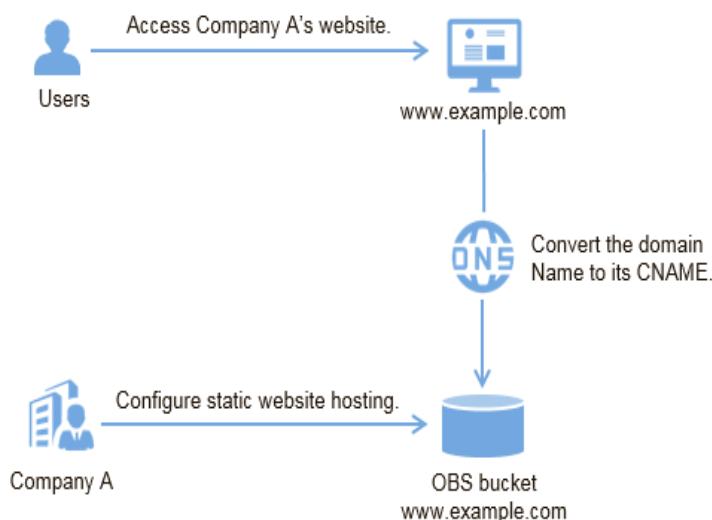
### 2.16.1 Static Website Hosting Overview

You can upload the content files of static websites to your bucket on OBS, authorize anonymous users the permission to read these files, and configure static website hosting for the bucket to host these files.

Static websites contain static web pages and some scripts that can run on clients, such as JavaScript and Flash. Different from static websites, dynamic websites rely on servers to process scripts, including PHP, JSP, and ASP.NET. OBS does not support scripts running on servers.

The configuration of static website hosting takes effect within two minutes. After the static website hosting is effective in OBS, you can access the static website by using the URL provided by OBS.

**Figure 2-81** Static website hosting



### 2.16.2 Redirection Overview

When using static website hosting, you can also configure redirection to redirect specific or all requests.

If the structure, address, or file name extension of a website is changed, users will fail to access the website using the old address (such as the address saved in the folder of favorites), and the 404 error message is returned. In this case, you can configure redirection for the website to redirect user access requests to the specified page instead of returning the 404 error page.

Typical configurations include:

- Redirecting all requests to another website.
- Redirecting specific requests based on redirection rules.

## 2.16.3 Configuring Static Website Hosting

You can configure static website hosting for a bucket and then use the bucket's domain name to access static websites hosted in the bucket.

The configuration of static website hosting takes two minutes at most to take effect.

### Prerequisites

Web page files required for static website hosting have been uploaded to the specified bucket.

The static website files hosted in the bucket are accessible to all users.

Static web page files in the Cold storage class have been restored. For more information, see [Restoring Objects from the Cold Storage](#).

### Procedure

**Step 1** In the bucket list, click the bucket you want to operate to go to the **Objects** page.

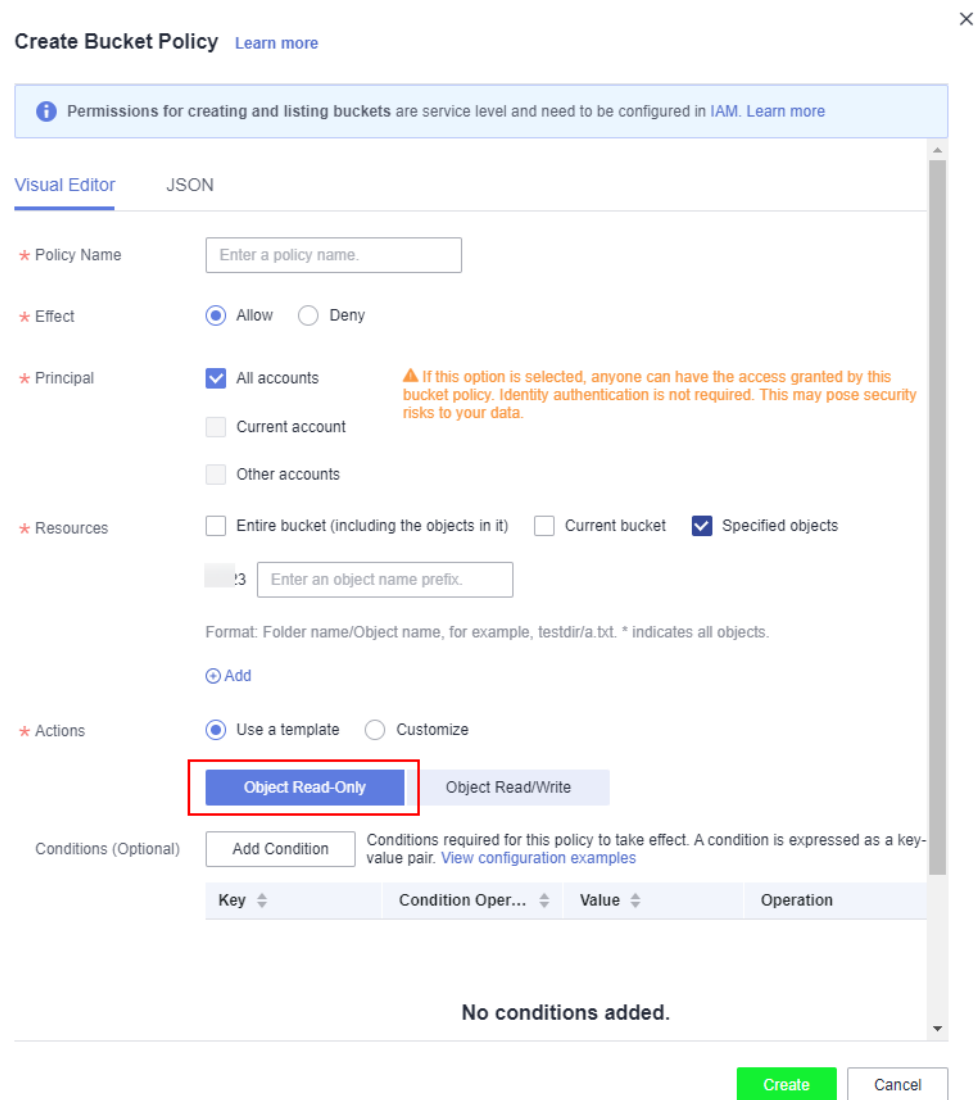
**Step 2 (Optional)** If the static website files in the bucket are not accessible to everyone, perform this step. If they are already accessible to everyone, skip this step.

To grant required permissions, see [Granting Anonymous Users Permission to Access Objects](#).

If the bucket contains only static website files, configure the **Object Read-Only** policy for the bucket, so that all files in it are publicly accessible.

1. Choose **Permissions > Bucket Policies**.
2. Click **Create**.
3. Configure bucket policy information.

**Figure 2-82** Granting the Object Read-Only permission



**Table 2-39** Parameters for configuring a public read policy

Parameter		Description
Configuration method		<b>Visual Editor</b> and <b>JSON</b> are available. Choose <b>Visual Editor</b> here. For details, see <a href="#">Creating a Custom Bucket Policy (JSON View)</a> .
Policy Name		Enter a custom policy name.
Policy content	Effect	Select <b>Allow</b> .
	Principals	Select <b>All accounts</b> .
	Resources	<ul style="list-style-type: none"> <li>- Select <b>Specified objects</b>.</li> <li>- Set the resource path to * (indicating all objects in the bucket).</li> </ul>



Parameter		Description
	Actions	<ul style="list-style-type: none"> <li>- Choose <b>Use a template</b>.</li> <li>- Select <b>Object Read-Only</b>.</li> </ul>

4. Click **Create**. The bucket policy is created.

**Step 3** In the navigation pane, choose **Overview**.

**Step 4** In the **Basic Configurations** area, click **Static Website Hosting**. The **Static Website Hosting** page is displayed.

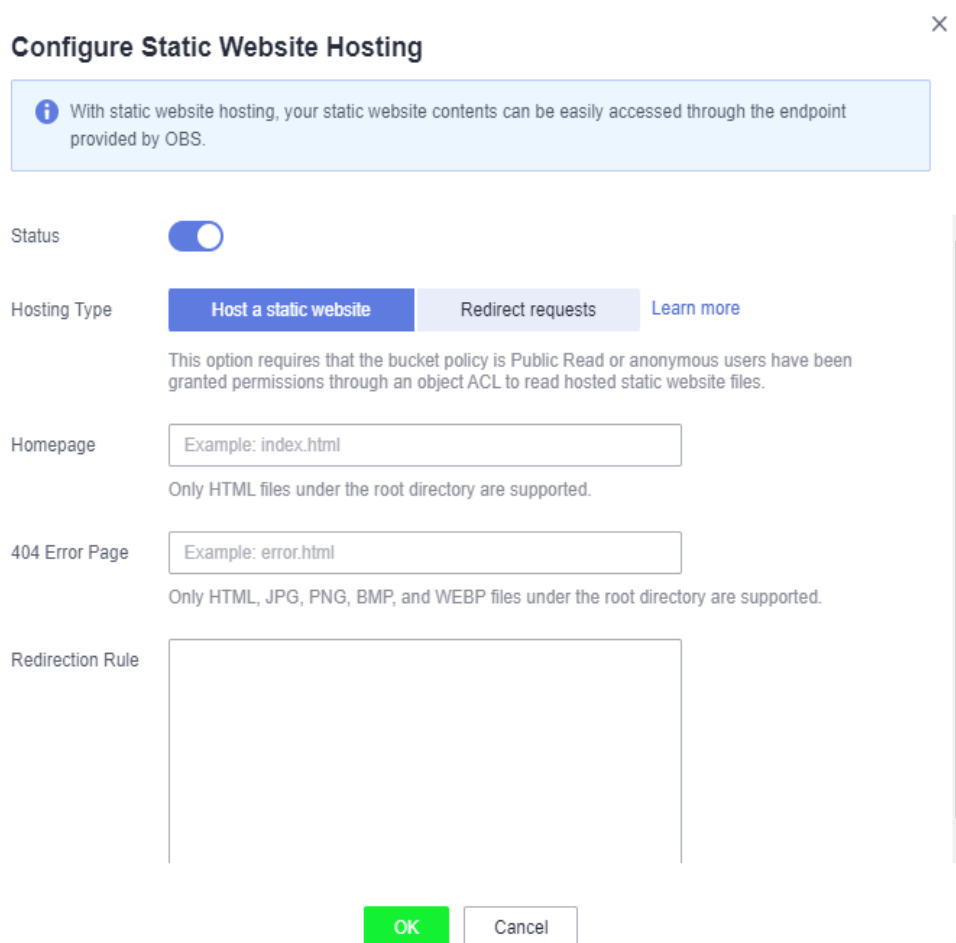
Alternatively, you can choose **Basic Configurations > Static Website Hosting** from the navigation pane on the left.

**Step 5** Click **Configure Static Website Hosting**. The **Configure Static Website Hosting** dialog box is displayed.

**Step 6** Enable **Status**.

**Step 7** Set the hosting type to the current bucket. For details, see [Figure 2-83](#).

**Figure 2-83** Configuring static website hosting



**Step 8** Configure the homepage and 404 error page.

- **Homepage:** specifies the default homepage of the static website. When OBS Console is used to configure static website hosting, only HTML web pages are supported. When APIs are used to configure static website hosting, OBS does not have any restriction but the **Content-Type** of objects must be specified.  
OBS only allows files such as **index.html** in the root directory of a bucket to function as the default homepage. Do not set the default homepage with a multi-level directory structure (for example, **/page/index.html**).
- **404 Error Page:** specifies the error page returned when an error occurs during static website access. When OBS Console is used to configure static website hosting, only HTML, JPG, PNG, BMP, and WebP files under the root directory are supported. When APIs are used to configure static website hosting, OBS does not have any restriction but the **Content-Type** of objects must be specified.

**Step 9 Optional:** In **Redirection Rules**, configure redirection rules. Requests that comply with the redirection rules are redirected to the specific host or page.

A redirection rule is compiled in the JSON or XML format. Each rule contains a **Condition** and a **Redirect**. The parameters are described in [Table 2-40](#).

**Table 2-40** Parameter description

Container	Key	Description
Condition	KeyPrefixEquals	Object name prefix on which the redirection rule takes effect. When a request is sent for accessing an object, the redirection rule takes effect if the object name prefix matches the value specified for this parameter.  For example, to redirect the request for object <b>ExamplePage.html</b> , set the <b>KeyPrefixEquals</b> to <b>ExamplePage.html</b> .
	HttpErrorCodeReturnedEquals	HTTP error codes upon which the redirection rule takes effect. The specified redirection is applied only when the error code returned equals the value specified for this parameter.  For example, if you want to redirect requests to <b>NotFound.html</b> when HTTP error code 404 is returned, set <b>HttpErrorCodeReturnedEquals</b> to <b>404</b> in <b>Condition</b> , and set <b>ReplaceKeyWith</b> to <b>NotFound.html</b> in <b>Redirect</b> .
Redirect	Protocol	Protocol used for redirecting requests. The value can be <b>http</b> or <b>https</b> . If this parameter is not specified, the default value <b>http</b> is used.

Container	Key	Description
	HostName	Host name to which the redirection is pointed. If this parameter is not specified, the request is redirected to the host from which the original request is initiated.
	ReplaceKeyPrefix- With	The object name prefix used in the redirection request. OBS replaces the value of <b>KeyPrefixEquals</b> with the value you specified here for <b>ReplaceKeyPrefixWith</b> . For example, to redirect requests for <b>docs</b> (objects in the <b>docs</b> directory) to <b>documents</b> (objects in the <b>documents</b> directory), set <b>KeyPrefixEquals</b> to <b>docs</b> under <b>Condition</b> and <b>ReplaceKeyPrefix- With</b> to <b>documents</b> under <b>Redirect</b> . This way, requests for object <b>docs/a.html</b> will be redirected to <b>documents/a.html</b> .
	ReplaceKeyWith	The object name used in the redirection request. OBS replaces the entire object name in the request with the value you specified here for <b>ReplaceKeyWith</b> . For example, to redirect requests for all objects in the <b>docs</b> directory to <b>documents/error.html</b> , set <b>KeyPrefixEquals</b> to <b>docs</b> under <b>Condition</b> and <b>ReplaceKeyWith</b> to <b>documents/ error.html</b> under <b>Redirect</b> . This way, requests for both objects <b>docs/a.html</b> and <b>docs/b.html</b> will be redirected to <b>documents/error.html</b> .
	HttpRedirectCode	HTTP status code returned to the redirection request. The default value is <b>301</b> , indicating that requests are permanently redirected to the location specified by <b>Redirect</b> . You can also set this parameter based on your service needs.

### Example of setting a redirection rule

- Example 1: All requests for objects prefixed with **folder1/** are automatically redirected to pages prefixed with **target.html** on host **www.example.com** using HTTPS.

```
[  
  {  
    "Condition": {  
      "KeyPrefixEquals": "folder1/"  
    },  
    "Redirect": {  
      "Protocol": "https",  
      "HostName": "www.example.com",
```

```
    "ReplaceKeyPrefixWith": "target.html"
  }
}
```

- Example 2: All requests for objects prefixed with **folder2/** are automatically redirected to objects prefixed with **folder/** in the same bucket.

```
[
  {
    "Condition": {
      "KeyPrefixEquals": "folder2/"
    },
    "Redirect": {
      "ReplaceKeyPrefixWith": "folder/"
    }
  }
]
```

- Example 3: All requests for objects prefixed with **folder.html** are automatically redirected to the **folderdeleted.html** object in the same bucket.

```
[
  {
    "Condition": {
      "KeyPrefixEquals": "folder.html"
    },
    "Redirect": {
      "ReplaceKeyWith": "folderdeleted.html"
    }
  }
]
```

- Example 4: If the HTTP status code 404 is returned, the request is automatically redirected to the page prefixed with **report-404/** on host **www.example.com**.

For example, if you request the page **ExamplePage.html** but the HTTP 404 error is returned, the request will be redirected to the **report-404/ExamplePage.html** page on the **www.example.com**. If the 404 redirection rule is not specified, the default 404 error page configured in the previous step is returned when the HTTP 404 error occurs.

```
[
  {
    "Condition": {
      "HttpErrorCodeReturnedEquals": "404"
    },
    "Redirect": {
      "HostName": "www.example.com",
      "ReplaceKeyPrefixWith": "report-404/"
    }
  }
]
```

#### Step 10 Click **OK**.

After the static website hosting is effective in OBS, you can access the static website by using the URL provided by OBS.

#### **NOTE**

In some conditions, you may need to clear the browser cache before the expected results are displayed.

----**End**

## 2.16.4 Configuring Redirection

You can redirect all requests for a bucket to another bucket or URL by configuring redirection rules.

### Prerequisites

Web page files required for static website hosting have been uploaded to the specified bucket.

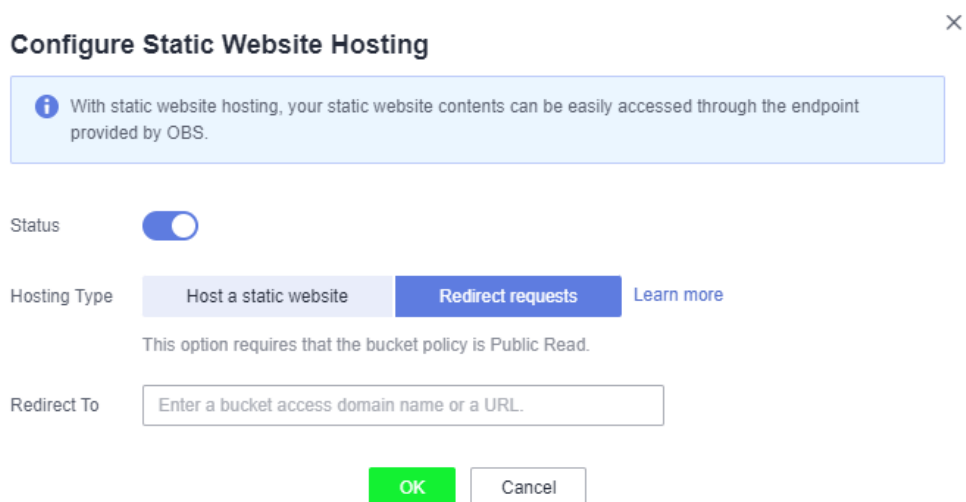
The static website files hosted in the bucket are accessible to all users.

Static web page files in the Cold storage class have been restored. For more information, see [Restoring Objects from the Cold Storage](#).

### Procedure

- Step 1** In the bucket list, click the bucket you want to operate to go to the **Objects** page.
- Step 2** In the navigation pane, choose **Overview**.
- Step 3** In the **Basic Configurations** area, click **Static Website Hosting**. The **Static Website Hosting** page is displayed.  
  
Alternatively, you can choose **Basic Configurations** > **Static Website Hosting** from the navigation pane on the left.
- Step 4** Click **Configure Static Website Hosting**. The **Configure Static Website Hosting** dialog box is displayed.
- Step 5** Enable **Status**.
- Step 6** Set **Hosting Type** to **Redirect requests**, as shown in [Figure 2-84](#). In the text box of **Redirect To**, enter the bucket's access domain name or URL.

**Figure 2-84** Configuring redirection



- Step 7** Click **OK**.
- Step 8** In the bucket list, click the bucket to which requests for the static website are redirected.

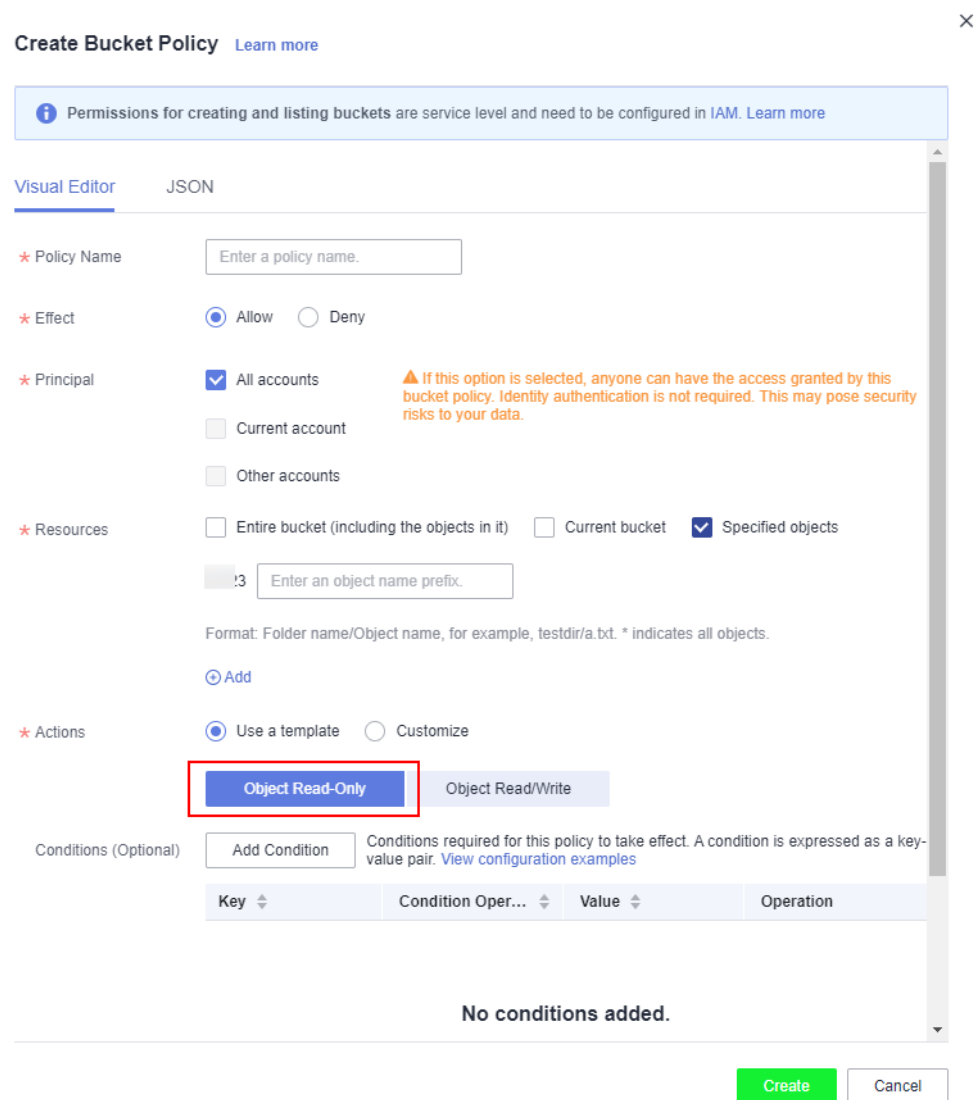
**Step 9 (Optional)** If the static website files in the bucket are not accessible to everyone, perform this step. If they are already accessible to everyone, skip this step.

To grant required permissions, see [Granting Anonymous Users Permission to Access Objects](#).

If the bucket contains only static website files, configure the **Object Read-Only** policy for the bucket, so that all files in it are publicly accessible.

1. Choose **Permissions > Bucket Policies**.
2. Click **Create**.
3. Configure bucket policy information.

**Figure 2-85** Granting the Object Read-Only permission



**Table 2-41** Parameters for configuring a public read policy

Parameter		Description
Configuration method		<b>Visual Editor</b> and <b>JSON</b> are available. Choose <b>Visual Editor</b> here. For details, see <a href="#">Creating a Custom Bucket Policy (JSON View)</a> .
Policy Name		Enter a custom policy name.
Policy content	Effect	Select <b>Allow</b> .
	Principals	Select <b>All accounts</b> .
	Resources	<ul style="list-style-type: none"><li>– Select <b>Specified objects</b>.</li><li>– Set the resource path to <b>*</b> (indicating all objects in the bucket).</li></ul>
	Actions	<ul style="list-style-type: none"><li>– Choose <b>Use a template</b>.</li><li>– Select <b>Object Read-Only</b>.</li></ul>

4. Click **Create**. The bucket policy is created.

**Step 10 Verification:** Input the access domain name of the bucket in the web browser and press **Enter**. The bucket or URL to which requests are redirected will be displayed.

 **NOTE**

In some conditions, you may need to clear the browser cache before the expected results are displayed.

----End

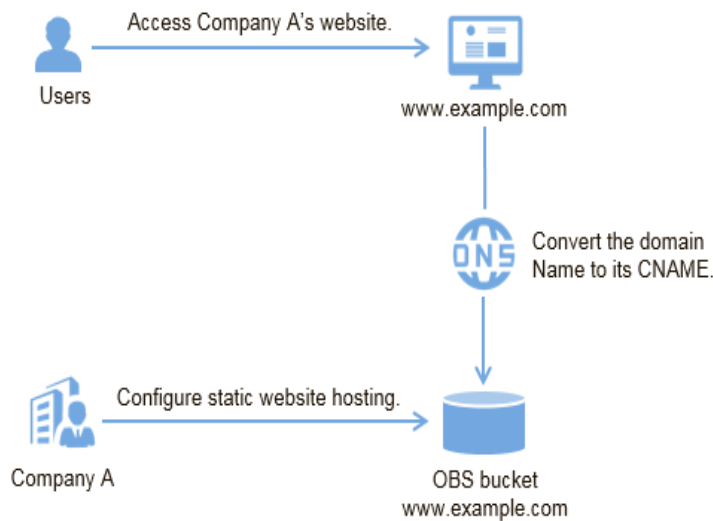
## 2.16.5 Using a User-Defined Domain Name to Configure Static Website Hosting

OBS allows you to access static websites hosted by OBS using user-defined domain names. This section uses a specific scenario as an example to describe how to use a user-defined domain name to configure static website hosting. For a basic understanding of the concepts and operations about the static website hosting on OBS, see [Configuring Static Website Hosting](#).

### Scenario

Company **A** has a large number of files to archive but it does not want to put the time and effort into its storage resources. Therefore, the company subscribes to OBS for hosting static websites and expects that the usernames under the company account can access the static resources through a user-defined domain name. See [Figure 2-86](#).

**Figure 2-86** Using a user-defined domain name to access hosted static website



## Operation Process

Create a bucket on OBS Console first, for storing static website resources, and enable static website hosting for this bucket. Then use DNS to create and configure domain name hosting. The procedure is as follows:

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

## Data Planning

**Table 2-42** describes the data to be planned before this configuration.

**Table 2-42** Data planning

Item	Description	Example
User-defined domain name	Indicates user's own domain name.	www.example.com
Static website homepage	Indicates the index page that is returned when you access a static website, that is, the homepage.	index.html



Item	Description	Example
404 error page	When an incorrect static website path is accessed, the 404 error page is returned.	error.html

- For example, the content of the **index.html** file is as follows:

```
<html>
<head>
  <title>Hello OBS!</title>
  <meta charset="utf-8">
</head>
<body>
  <p>Welcome to use OBS static website hosting.</p>
  <p>This is the homepage.</p>
</body>
</html>
```

- For example, the content of the **error.html** file is as follows:

```
<html>
<head>
  <title>Hello OBS!</title>
  <meta charset="utf-8">
</head>
<body>
  <p>Welcome to use OBS static website hosting.</p>
  <p>This is the 404 error page.</p>
</body>
</html>
```

## 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 of your choice. 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 are no special requirements on bucket names. Create a bucket for storing static website files as prompted. The following example describes how to create a bucket named **example**:

1. Log in to OBS Console.
2. Click **Create Bucket** in the upper right corner of the page.
3. Configure the following parameters in the dialog box that is displayed:
  - **Region**: Select a region closest to you.
  - **Bucket Name**: Enter **example**.
  - **Storage Class**: It is recommended that you select **Standard**.

#### NOTE

You can also select the Warm, or Cold storage class based on the website requirements for access frequency and speed. For details about storage classes, see [Storage Classes Overview](#).

- **Bucket Policy:** Select **Public Read** to allow any user to access objects in the bucket.
  - **Server-Side Encryption:** Select **Disable**.
4. Click **Create Now** to complete the creation.

### Step 3 Upload static website files to the bucket.

Prepare the static website files to be uploaded and perform the following steps to upload all static website files to bucket **example**.

1. Click the bucket name **example** to go to the **Objects** page.
2. Click **Upload Object**. A dialog box shown in **Figure 2-87** is displayed.

**Figure 2-87** Uploading objects



3. Drag the prepared static website files to the **Upload Object** area. You can also click **add files** in the **Upload Object** area to select files.

#### **NOTE**

- The static website files cannot be encrypted for upload.
  - The website home page file (**index.html**) and 404 error page (**error.html**) must be stored in the root directory of the bucket.
  - It is recommended that you select **Standard** for the storage class. If the storage class of a static website file is Cold, you need to restore the static website file before you can access it. For details, see **Restoring Objects from the Cold Storage**.
4. Click **Upload** to complete the upload.

### Step 4 Configure static website hosting.

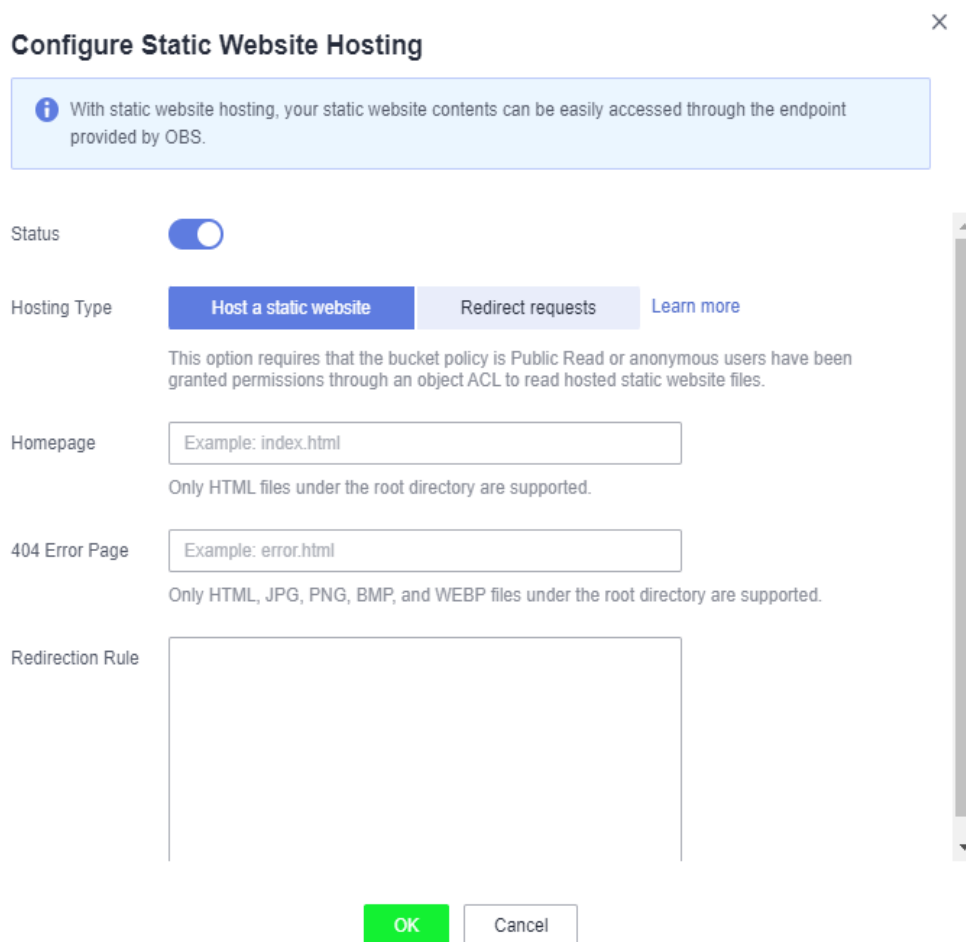
After uploading the static website files, you need to configure the static website hosting function for the bucket.

 NOTE

You can also redirect the entire static website to another bucket or domain name. For details, see [Configuring Redirection](#).

1. Click the bucket name **example** to go to the **Objects** page.
2. In the navigation pane, choose **Basic Configurations** > **Static Website Hosting**. The **Static Website Hosting** page is displayed.
3. Click **Configure Static Website Hosting** to open the dialog box.
4. Enable **Status**.
5. Set **Hosting Type** to **Host a static website**.

**Figure 2-88** Configuring static website hosting



 NOTE

You can also configure redirection rules based on service requirements to implement website content redirection. For details, see [Configuring Static Website Hosting](#).

6. Set the **Home Page** to **index.html** as planned, and the **404 Error Page** to **error.html**.
7. Click **OK**.

**Step 5** Bind a user-defined domain name.

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

1. Click the bucket name **example** to go to the **Objects** page. In the navigation pane, choose **Domain Name Mgmt.**
2. Click **Configure User Domain Name** and set **User Domain Name** to **www.example.com**.

**Figure 2-89** Configuring a user domain name

**Configure User Domain Name** ×

**Add User Domain Name**

1 Add a domain name so that you can use it to access the files stored in the bucket.

Resolve CNAME

2 Resolve the domain name to finish the binding.

**i** The bound user domain names only support access over HTTP now.

\* Origin Server  u □

\* User Domain Name

(1/5)

3. Click **OK**. The user-defined domain name is bound to the bucket.

**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, directly manage your user-defined domain names on 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.

Alternatively, you can add a CNAME record to the DNS at the DNS registrar, mapping to the static website domain name hosted by the bucket.

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 about how to create a public zone, see "Step 1. Create a Public Zone" in section "Routing Internet Traffic to a Website" of the *Domain Name Service User Guide*.
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 to map, that is, the static website domain name hosted by bucket **example**.

For details, see section "Adding a CNAME Record Set" in the *Domain Name Service User Guide*.

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 addresses of the DNS servers, see "Step 4. Change DNS Servers of the Domain Name" in section "Routing Internet Traffic to a Website" of the *Domain Name Service User Guide*.

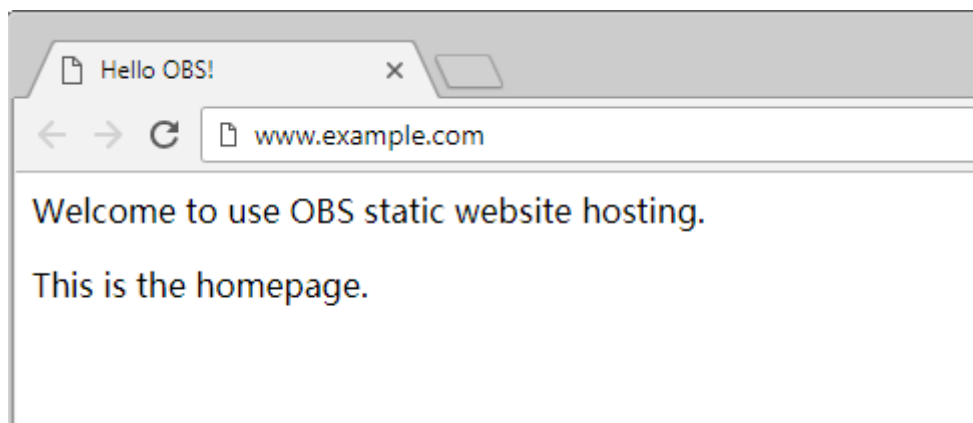
#### NOTE

The address change will be effective within 48 hours. The actual time taken varies depending on the domain name registrar.

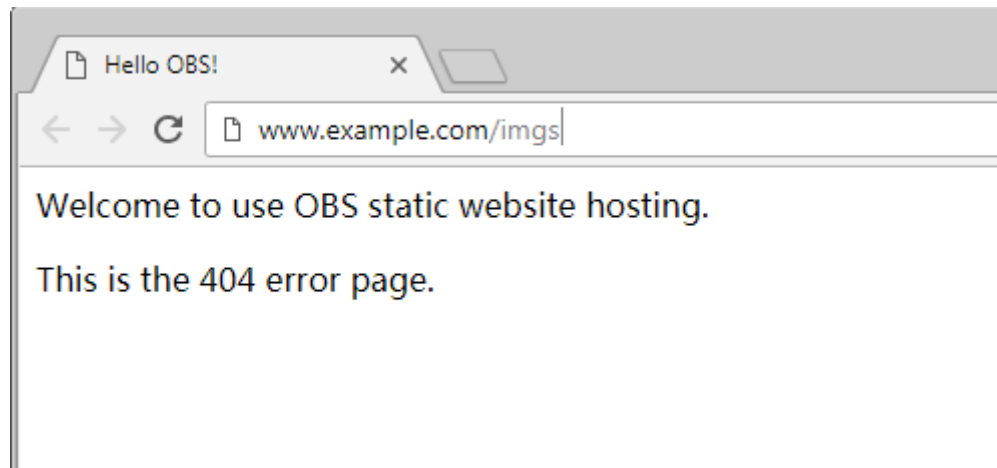
#### **Step 7** Verify that the configuration is successful.

- Enter the following URL in the address box of the browser: **www.example.com**, to check whether the default homepage can be accessed. See [Figure 2-90](#).

**Figure 2-90** Default homepage



- In the web browser, enter a static file access address that does not exist in a bucket. For example, enter **www.example.com/imgs** to verify that the 404 error page (error.html) can be returned. [Figure 2-91](#) displays the error page.

**Figure 2-91** 404 error page**NOTE**

In some conditions, you may need to clear the browser cache before the expected results are displayed.

----End

## Website Update

If you need to update a static file, such as a picture, a piece of music, an HTML file, or a CSS file, you can re-upload the static file.

By default, if two files in a path share one name, the newly uploaded file overwrites the original one. To prevent files from being overwritten, you can enable the versioning function. Versioning allows you to keep multiple versions of a static file, so that you can retrieve and restore history versions conveniently. With versioning enabled, data can be restored rapidly when accidental operations or application faults occur. For detailed information about versioning, see chapter [Versioning Overview](#).

## 2.17 Cross-Origin Resource Sharing

### 2.17.1 CORS Overview

CORS is a browser-standard mechanism provided by the World Wide Web Consortium (W3C). It defines the interaction methods between client-side web applications in one origin and resources in another origin. For general web page requests, website scripts and contents in one origin cannot interact with those in another origin because of Same Origin Policies (SOPs).

The CORS specification is supported to allow cross-origin requests to access OBS resources.

OBS supports static website hosting. Static websites stored in OBS can respond to website requests from another origin only when CORS is configured for the bucket.

Typical application scenarios of CORS are as follows:

- Enables JavaScript and HTML5 to be used for establishing web applications that can directly access resources in OBS. No proxy servers are required for transfer.
- Enables the dragging function of HTML5 to be used to upload files to OBS (with the upload progress displayed) or update OBS contents using web applications.
- Hosts external web pages, style sheets, and HTML5 applications in different origins. Web fonts or pictures in OBS can be shared by multiple websites.

The configuration of CORS takes effect within two minutes.

## 2.17.2 Configuring CORS

This section describes how to use CORS in HTML5 to implement cross-origin access.

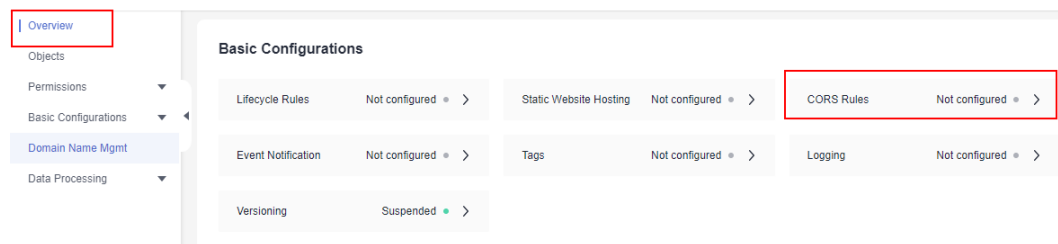
### Prerequisites

Static website hosting has been configured. For details, see [Configuring Static Website Hosting](#).

### Procedure

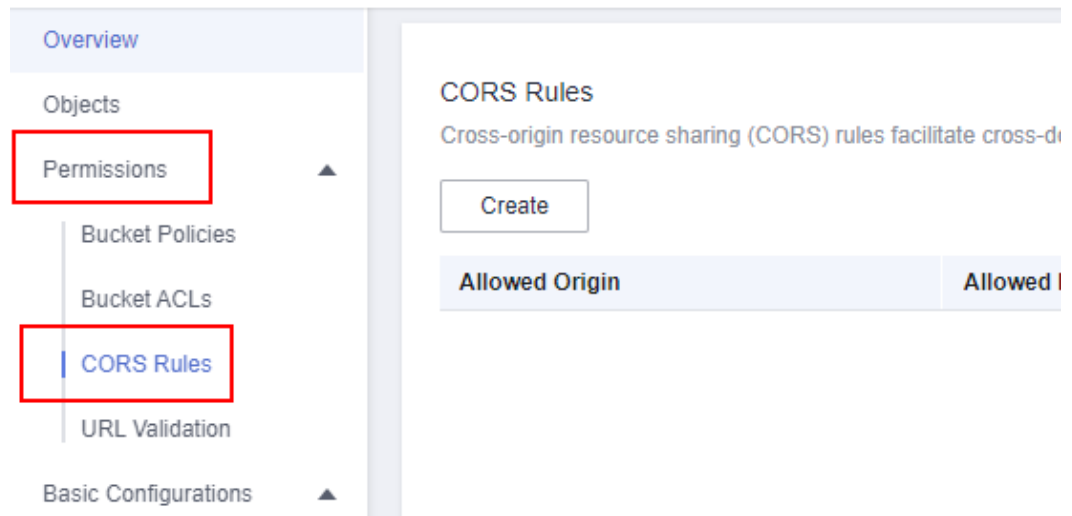
- Step 1** In the bucket list, click the bucket you want to operate to go to the **Objects** page.
- Step 2** In the navigation pane, choose **Overview**.
- Step 3** In the **Basic Configurations** area, click **CORS Rules**. The **CORS Rules** page is displayed.

**Figure 2-92** Overview > Basic Configurations > CORS Rules



Alternatively, you can choose **Permissions > CORS Rules** in the navigation pane.

**Figure 2-93** Permissions > CORS Rules

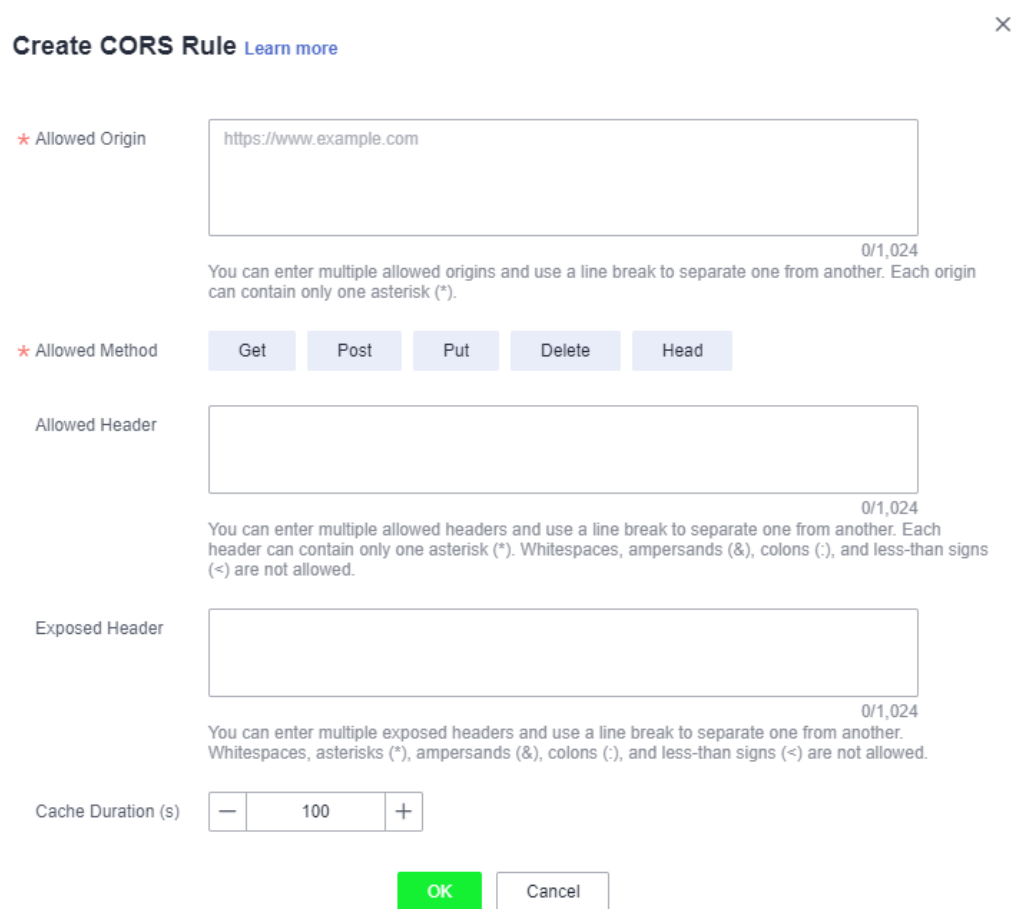


**Step 4** Click **Create**. The **Create CORS Rule** dialog box is displayed. See [Figure 2-94](#) for details.

**NOTE**

A bucket can have a maximum of 100 CORS rules configured.

**Figure 2-94** Creating a CORS rule





**Step 5** In the **CORS Rule** dialog box, configure **Allowed Origin**, **Allowed Method**, **Allowed Header**, **Exposed Header**, and **Cache Duration (s)**.

**Table 2-43** Parameters in CORS rules

Parameter	Description
Allowed Origin	<p>Mandatory</p> <p>Specifies the origins from which requests can access the bucket.</p> <p>Multiple matching rules are allowed. One rule occupies one line, and allows one wildcard character (*) at most. An example is given as follows:</p> <pre>http://rds.example.com https://*.vbs.example.com</pre>
Allowed Method	<p>Mandatory</p> <p>Specifies the allowed request methods for buckets and objects.</p> <p>The methods include Get, Post, Put, Delete, and Head.</p>
Allowed Header	<p>Optional</p> <p>Specifies the allowed headers in cross-origin requests. Only CORS requests matching the allowed headers are valid.</p> <p>You can enter multiple allowed headers (one per line) and each line can contain one wildcard character (*) at most. Spaces and special characters including &amp;&lt; are not allowed.</p>
Exposed Header	<p>Optional</p> <p>Specifies the exposed headers in CORS responses, providing additional information for clients.</p> <p>By default, a browser can access only headers <b>Content-Length</b> and <b>Content-Type</b>. If the browser wants to access other headers, you need to configure those headers in this parameter.</p> <p>You can enter multiple exposed headers (one per line). Spaces and special characters including *&lt; are not allowed.</p>
Cache Duration (s)	<p>Mandatory</p> <p>Specifies the duration that your browser can cache CORS responses, expressed in seconds. The default value is <b>100</b>.</p>

**Step 6** Click **OK**.

Message "The CORS rule created successfully." is displayed. The CORS configuration takes effect within two minutes.

After CORS is successfully configured, only the addresses specified in **Allowed Origin** can access a bucket in OBS using the methods specified in **Allowed Method**. For example, you can configure CORS parameters for bucket **testbucket** as follows:

- **Allowed Origin:** <https://www.example.com>
- **Allowed Method:** GET
- **Allowed Header:** \*
- **Exposed Header:** \*
- **Cache Duration (s):** 100

By doing so, OBS only allows GET requests from <https://www.example.com> to access bucket **testbucket**, without restrictions on request headers. The client can cache CORS responses for 100 seconds.

----End

## 2.18 URL Validation

### 2.18.1 URL Validation Overview

To reduce costs, some websites steal links from other websites to enrich their own contents. Link stealing not only damages interests of the original websites but also increases workloads on the original websites' servers. Therefore URL is used to resolve this problem.

In HTTP, a website can detect the web page that accesses a target web page using the **Referer** field. As the **Referer** field can trace sources, specific techniques can be used to block or return to specific web pages if the pages are not from the website. URL validation checks whether the **Referer** field in requests matches the whitelist or blacklist by setting **Referers**. If the field matches the whitelist, the requests are allowed. Otherwise, the requests are blocked or specific pages are displayed.

OBS supports URL validation based on the **Referer** header field in HTTP requests to prevent a user's data in OBS from being stolen by other users. OBS supports both whitelists and blacklists.

### 2.18.2 Configuring URL Validation

OBS blocks access requests from blacklisted URLs and allows those from whitelisted URLs.

#### Prerequisites

Static website hosting has been enabled.

#### Procedure

**Step 1** In the bucket list, click the bucket you want to operate to go to the **Objects** page.

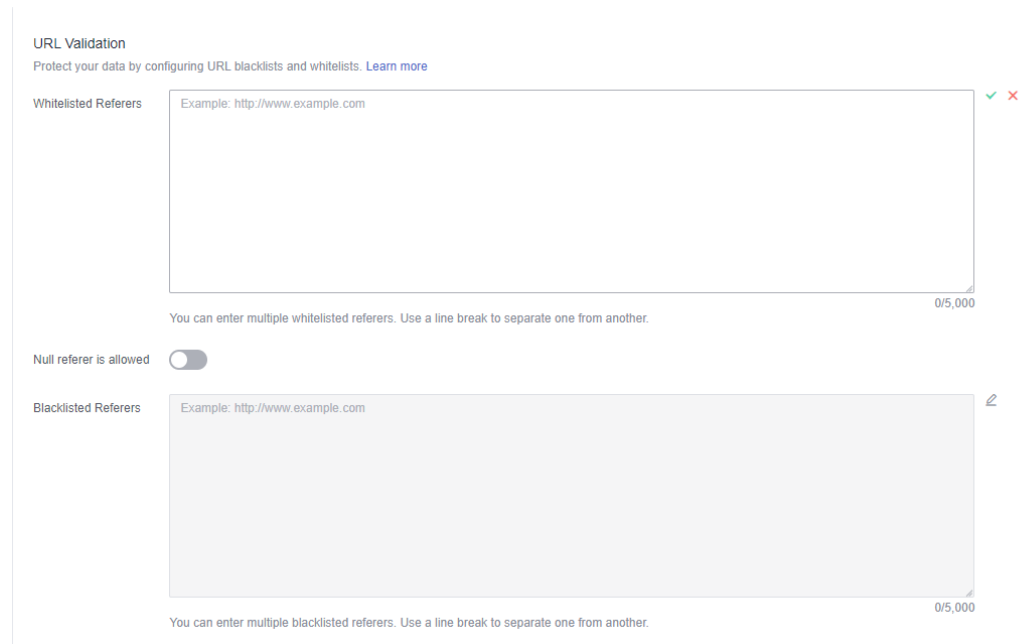
**Step 2** In the navigation pane, choose **Overview**.

**Step 3** In the **Basic Configurations** area, click **URL Validation**. The **URL Validation** page is displayed.

Alternatively, you can choose **Permissions > URL Validation** in the navigation pane on the left.

**Step 4** Click  next to the text box of **Whitelisted Referers** or **Blacklisted Referers**, and enter the referers.

**Figure 2-95** Configuring URL validation




#### Principles for setting **Referers**:

- The length of a whitelist or blacklist cannot exceed 1024 characters.
- Referer format:
  - You can enter multiple referers, each in a line.
  - The referer parameter supports asterisks (\*) and question marks (?). An asterisk works as a wildcard that can replace zero or multiple characters, and a question mark (?) can replace a single character.
  - If the referer header field contains **http** or **https** during download, the referer must contain **http** or **https**.
- If **Whitelisted Referers** is left blank but **Blacklisted Referers** is not, all websites except those specified in the blacklist are allowed to access data in the target bucket.
- If **Whitelisted Referers** is not left blank, only the websites specified in the whitelist are allowed to access the target bucket no matter whether **Blacklisted Referers** is left blank or not.

#### **NOTE**

If **Whitelisted Referers** is configured the same as **Blacklisted Referers**, the blacklist takes effect. For example, if both **Whitelisted Referers** and **Blacklisted Referers** are set to **https://www.example.com**, access requests from this address will be blocked.

- If **Whitelisted Referers** and **Blacklisted Referers** are both left blank, all websites are allowed to access data in the target bucket by default.
- Before determining whether a user has the four types of permissions (read, write, ACL read, and ACL write) for a bucket or objects in the bucket, check whether this user complies with the URL validation principles of the **Referer** field.

**Step 5** Click  to save the settings.

----End

## 2.19 Monitoring

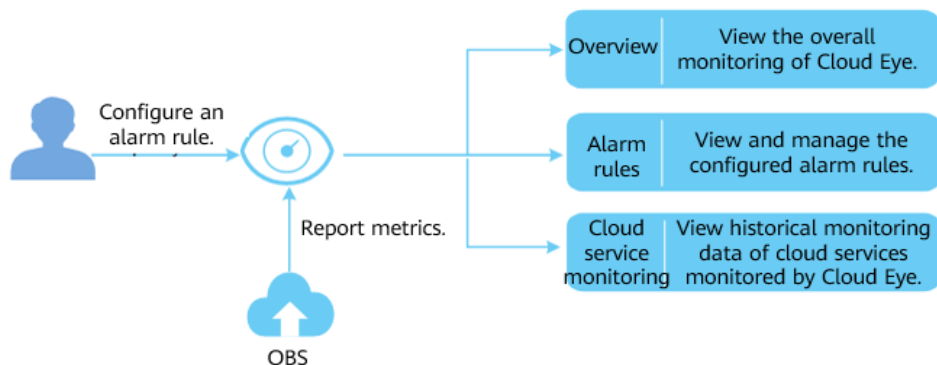
### 2.19.1 Monitoring OBS

#### Scenarios

In the use of OBS, you may send PUT and GET requests that generate upload and download traffic, or receive error responses from the server. To learn the requests, traffic, and error responses in a timely manner, you can use Cloud Eye to perform automatic and real-time monitoring over your buckets.

You do not need to separately subscribe to Cloud Eye. It starts automatically once you create a resource (a bucket, for example) in OBS. For more information about Cloud Eye, see *Cloud Eye User Guide*.

**Figure 2-96** Cloud Eye monitoring



#### Setting Alarm Rules

In addition to automatic and real-time monitoring, you can configure alarm rules in Cloud Eye to receive alarm notifications when specified events happen.

For details, see section "Creating an Alarm Rule" in *Cloud Eye User Guide*.

#### Viewing OBS Monitoring Metrics

Cloud Eye monitors **OBS monitoring metrics** in real time. You can view detailed monitoring statistics of each metric on the console of Cloud Eye.

For details, see section "Querying Cloud Service Monitoring Metrics" in *Cloud Eye User Guide*.

## 2.19.2 OBS Monitoring Metrics

### Functions

This section defines the namespace, list, and dimensions of monitoring metrics reported by OBS to Cloud Eye. You can use the management console or APIs provided by Cloud Eye to search for monitoring metrics and alarms generated by OBS.

### Namespace

SYS.OBS

### Monitoring Metrics

Metric ID	Metric	Description	Value Range	Monitored Entity	Monitoring Period (Original Metric)
download_bytes	Bytes Downloaded	Specifies the response bytes of all download requests made to all buckets in a region, including bytes in HTTP entity bodies. Unit: byte	≥ 0 bytes	Bucket	5 min
upload_bytes	Bytes Uploaded	Specifies the bytes of all upload requests made to all buckets in a region, including bytes in HTTP entity bodies. Unit: byte	≥ 0 bytes	Bucket	5 min
get_request_count	GET Requests	Specifies the number of GET, HEAD, or OPTIONS requests made to all buckets and objects in the buckets of a region. Unit: count	≥ 0 counts	Bucket	5 min

Metric ID	Metric	Description	Value Range	Monitored Entity	Monitoring Period (Original Metric)
put_request_count	PUT Requests	Specifies the number of PUT, POST, and DELETE requests made to all buckets and objects in the buckets of a region. Unit: count	≥ 0 counts	Bucket	5 min
first_byte_latency	First Byte Download Delay	Specifies the average time from receiving a GET, HEAD, or OPTIONS request to the time that the system starts to respond in a measurement period. Unit: ms	≥ 0 ms	Bucket	5 min
request_count_4xx	4xx Errors	Specifies the times that the server responds to requests whose error codes are 4xx. Unit: count	≥ 0 counts	Bucket	5 min
request_count_5xx	5xx Errors	Specifies the times that the server responds to requests whose error codes are 5xx. Unit: count	≥ 0 counts	Bucket	5 min

## Dimensions

Table 2-44 Dimensions

Key	Value
bucket_name	Bucket dimension. The value is the bucket name.

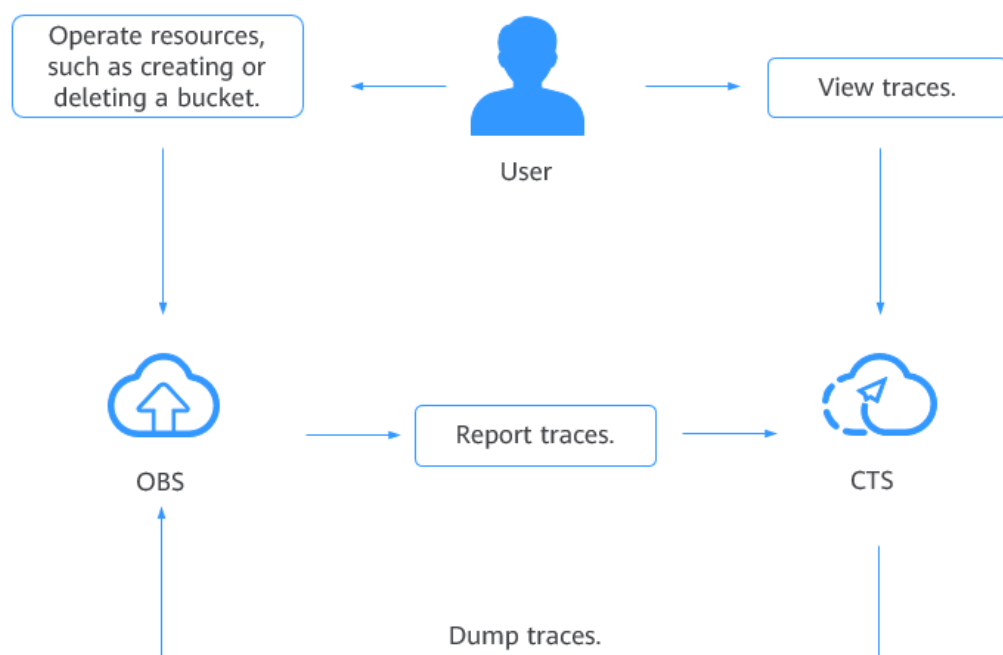
## 2.20 Cloud Trace Service

Cloud Trace Service (CTS) records operations on cloud resources in your account. You can use the records to perform security analysis, track resource changes, audit compliance, and locate faults.


After you enable CTS and configure a tracker, CTS can record management and data traces of OBS for auditing.

For details about how to enable and configure CTS, see [Getting Started](#).

**Figure 2-97** CTS



### Procedure

- Step 1** Log in to the management console.
- Step 2** In the upper left corner of the top navigation menu, click  to select a region.
- Step 3** Then choose **Service List > Management & Deployment > Cloud Trace Service**. The **Trace List** page is displayed.
- Step 4** Configure the cloud audit for OBS by referring to [Configuring a Tracker](#) in the *Cloud Trace Service User Guide*.

----End

**Table 2-45** OBS management operations logged by CTS

Tracker Type	Operation	Resource	Trace Name
Management	Deleting a bucket	bucket	deleteBucket
Management	Deleting the CORS configuration of a bucket	bucket	deleteBucketCors
Management	Deleting the custom domain name configuration	bucket	deleteBucketCustom-domain
Management	Deleting the lifecycle configuration of a bucket	bucket	deleteBucketLifecycle
Management	Deleting a bucket policy	bucket	deleteBucketPolicy
Management	Deleting the tag configuration of a bucket	bucket	deleteBucketTagging
Management	Deleting the static website hosting configuration of a bucket	bucket	deleteBucketWebsite
Management	Creating a bucket	bucket	createBucket
Management	Configuring the bucket ACL	bucket	setBucketAcl
Management	Configuring the CORS rule for a bucket	bucket	setBucketCors
Management	Setting the custom domain name for a bucket	bucket	setBucketCustomdomain
Management	Configuring the bucket lifecycle rules	bucket	setBucketLifecycle
Management	Configuring the bucket logging function	bucket	setBucketLogging
Management	Configuring the event notification function for buckets	bucket	setBucketNotification



Tracker Type	Operation	Resource	Trace Name
Management	Configuring the bucket policy	bucket	setBucketPolicy
Management	Configuring the bucket quota	bucket	setBucketQuota
Management	Configuring the bucket storage class	bucket	setBucketStorageclass
Management	Configuring the bucket tag	bucket	setBucketTagging
Management	Configuring the versioning function for buckets	bucket	setBucketVersioning
Management	Configuring the static domain name for buckets	bucket	setBucketWebsite
Management	Configuring bucket's default encryption	bucket	setBucketEncryption
Management	Deleting bucket's default encryption settings	bucket	deleteBucketEncryption

**Table 2-46** OBS data operations logged by CTS

Tracker Type	Operation	Resource	Trace Name
Data_Read	Downloading an object	object	GET.OBJECT
Data_Read	Querying the object ACL	object	GET.OBJECT.ACL
Data_Read	Querying the bucket website configuration	object	GET.OBJECT.WEBSITE
Data_Read	Accessing an object through the website	object	HEAD.OBJECT.WEBSITE
Data_Read	Querying the object metadata	object	HEAD.OBJECT
Data_Read	Listing part data	object	LIST.OBJECT.UPLOAD

Tracker Type	Operation	Resource	Trace Name
Data_Write	Deleting an object	object	DELETE.OBJECT
Data_Write	Canceling a part	object	DELETE.UPLOAD
Data_Write	Queries the cross-domain requests for objects	object	OPTIONS.OBJECT
Data_Write	Uploading an object	object	POST.OBJECT
Data_Write	Deleting objects in batches	object	POST.OBJECT.MULTIDELETE
Data_Write	Restoring Cold objects	object	POST.OBJECT.RESTORE
Data_Write	Merging parts	object	POST.UPLOAD.COMPLET E
Data_Write	Initializing multipart tasks	object	POST.UPLOAD.INIT
Data_Write	Uploading an object	object	PUT.OBJECT
Data_Write	Configuring the object ACL	object	PUT.OBJECT.ACL
Data_Write	Copying an object	object	PUT.OBJECT.COPY
Data_Write	Configuring the object storage class	object	PUT.OBJECT.STORAGECL ASS
Data_Write	Uploading a part	object	PUT.PART
Data_Write	Copying a part	object	PUT.PART.COPY

## Follow-up Procedure

You can click **Disable** under the **Operation** column on the right of a tracker to disable the tracker. After the tracker is disabled, the system will stop recording operations, but you can still view existing operation records.

You can click **Delete** under the **Operation** column on the right of a tracker to delete the tracker. Deleting a tracker has no impact on existing operation records. When you enable CTS again, you can view operation records that have been generated.

## 2.21 Task Center

When you upload objects, restore objects in batches, change storage classes in batches, or delete folders, corresponding records of the tasks will be displayed in the task center for you to view the tasks' progress and status.

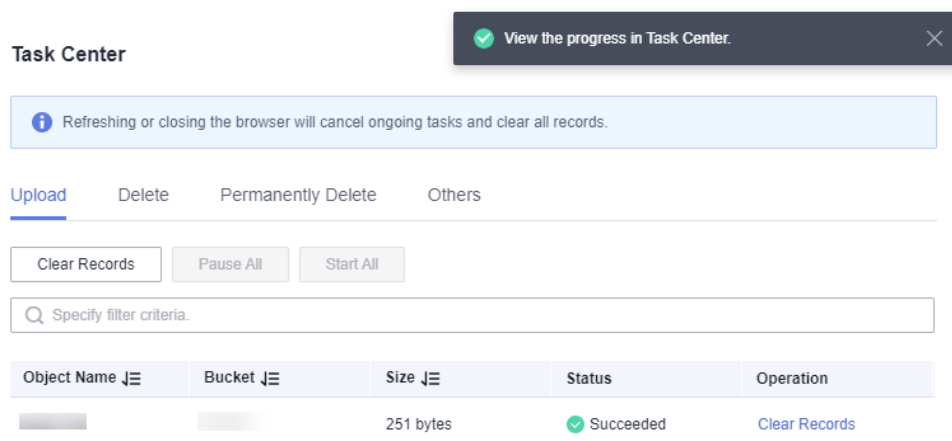
### NOTE

Refreshing or closing the browser will cancel ongoing tasks and clear all records.

### Procedure

- Step 1** In the object list of your bucket, click **Task Center** in the upper right corner.
- Step 2** View the records of uploading objects, restoring objects in batches, changing storage classes in batches, or deleting folders.
  - Click **Clear Records** to clear all task records.
  - On the **Upload** tab page, you can click **Pause All** or **Start All** to manage upload tasks in batches.

Figure 2-98 Task Center



----End

## 2.22 Related Operations

### 2.22.1 Creating an IAM Agency

To use some OBS features, you need to use IAM agencies to grant required permissions to OBS for processing your data.

#### Creating an Agency for Uploading Logs

- Step 1** In the **Logging** dialog box, click **Create Agency** to jump to the **Agencies** page on the **Identity and Access Management** console.

**Step 2** Click **Create Agency**.

**Step 3** Enter an agency name.

**Step 4** Select **Cloud service** for the **Agency Type**.

**Step 5** Select **Object Storage Service (OBS)** for **Cloud Service**.

**Step 6** Set a validity period.

**Step 7** Click **Next**.

**Step 8** On the **Select Policy/Role** page, select a custom policy that has the permission to upload data to the log storage bucket and click **Next**.

If no custom policy is available, create one by choosing **Permissions > Policies/Roles** in the navigation pane

Select **JSON** for **Policy View**. The policy content is as follows.

 **NOTE**

When coding the policy content in an actual scenario, replace **mybucketlogs** with the actual bucket name:

```
{
  "Version": "1.1",
  "Statement": [
    {
      "Action": [
        "obs:object:PutObject"
      ],
      "Resource": [
        "OBS:*:object:mybucketlogs/*"
      ],
      "Effect": "Allow"
    }
  ]
}
```

**Step 9** On the **Select Scope** page, select **Global services** for **Scope** and click **OK**.

**Step 10** (Optional) If the default encryption is enabled for the log storing bucket, the IAM agency also requires the **KMS Administrator** permission in the region where the log storing bucket resides.

1. Go to the **Agencies** page on the **Identity and Access Management** console and click the name of the agency created in the previous step.
2. Choose the **Permissions** tab and click **Authorize**.
3. On the **Select Policy/Role** page, search for and select **KMS Administrator**. Then, click **Next**.
4. On the **Select Scope** page, select **Region-specific projects** and then select the projects in the region where the log bucket resides.

----End

## 2.23 Troubleshooting

## 2.23.1 An Object Fails to Be Downloaded Using Internet Explorer 11

### Symptom

A user logs in to OBS Console using Internet Explorer 11 and uploads an object. When the user attempts to download the object to the original path to replace the original object without closing the browser, a message is displayed indicating a download failure. Why does this happen?

For example, a user uploads object **abc** from the root directory of local drive C to a bucket in OBS Console. When the user attempts to download the object to the root directory of local drive C to replace the original object without closing the browser, a message is displayed indicating a download failure.

### Answer

This problem is caused by browser incompatibility. It can be solved by using a different web browser.

If this problem occurs, close the browser and try again.

## 2.23.2 OBS Console Cannot Be Opened in Internet Explorer 9

### Question

Why OBS Console cannot be opened in Internet Explorer 9, even if the address of OBS Console can be pinged?

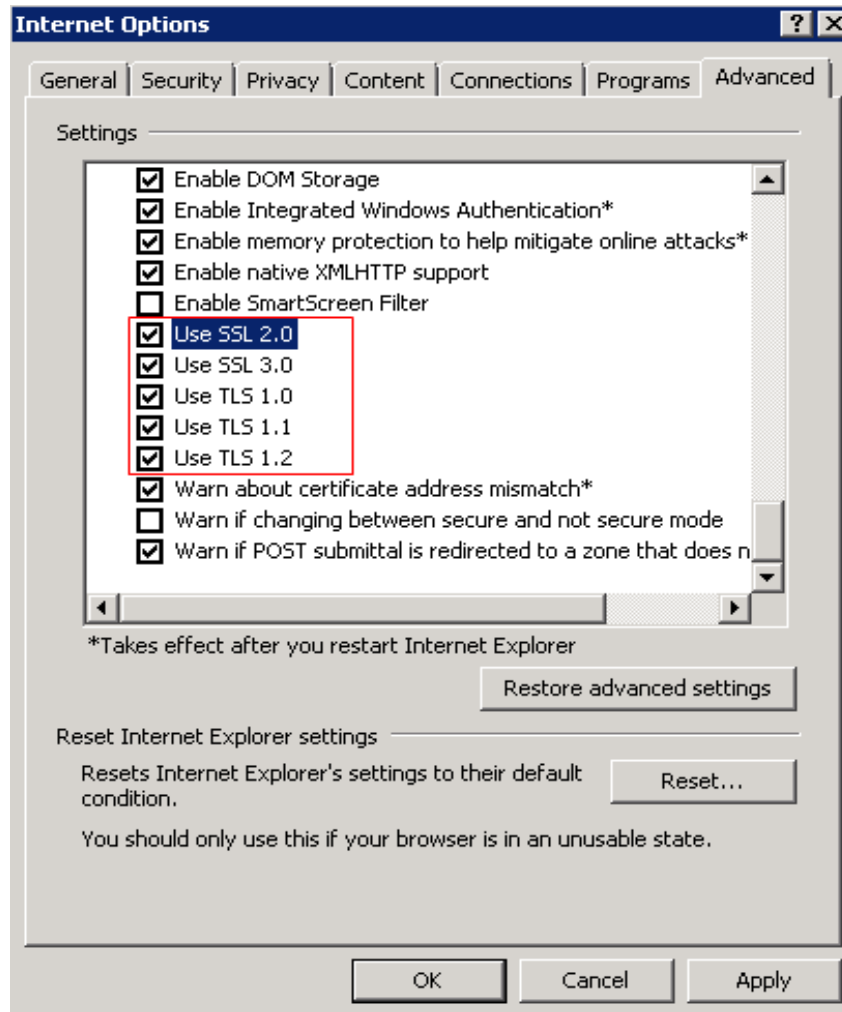
### Answer

Confirm whether **Use SSL** and **Use TLS** are selected in **Internet Options**. If not, perform the following procedure and try again:

**Step 1** Open Internet Explorer 9.

**Step 2** Click **Tools** in the upper right corner and choose **Internet Options > Advanced**. Then select **Use SSL 2.0**, **Use SSL 3.0**, **Use TLS 1.0**, **Use TLS 1.1**, and **Use TLS 1.2**, as shown in [Figure 2-99](#).

Figure 2-99 Internet Options



Step 3 Click OK.

----End

### 2.23.3 The Object Name Changes After an Object with a Long Name Is Downloaded to a Local Computer

#### Question

After an object with a relatively long name is downloaded to a local path, the object name changes.

#### Answer

For Windows, a file name, including the file name extension, can contain a maximum of 255 characters. When an object with a name containing more than 255 characters is downloaded to a local computer, the system keeps only the first 255 characters automatically.

## 2.23.4 Failed to Configure Event Notifications

### Question

During the configuration of event notifications on OBS, message "OBS is not authorized to use this topic. Go to SMN to authorize OBS to use this topic." is displayed.

### Answer

Go to the SMN console. On the **Configure Topic Policy** page, select **OBS** under **Services that can publish messages to this topic**.

For details about how to use the SMN service, see "Topic Policy" in the *SMN User Guide*.

## 2.23.5 Time Difference Is Longer Than 15 Minutes Between the Client and Server

### Question

Error message "Time difference is longer than 15 minutes between the client and server" or "The difference between the request time and the current time is too large" is displayed during the use of OBS.

### Answer

For security purposes, OBS verifies the time offset between the client and server. If the offset is longer than 15 minutes, the OBS server will reject your requests and this error message is reported. To resolve this problem, adjust your local time (UTC) and try again.

## 2.24 Error Code List

If a request fails to be processed due to errors, an error response is returned. An error response contains an error code and error details. [Table 2-47](#) lists some common error codes in OBS error responses.

**Table 2-47** OBS error codes

Error Code	Description
Obs.0000	Invalid parameter.
Obs.0001	All access requests to this object are invalid.
Obs.0002	The absolute path of a file cannot exceed 1023 characters. Please retry.
Obs.0003	The connection timed out.

Error Code	Description
Obs.0004	<p>Time difference is longer than 15 minutes between the client and server. Correctly set the local time.</p> <p>For security purposes, OBS verifies the time offset between the client and server. If the offset is longer than 15 minutes, the OBS server will reject your requests and this error message is reported. To resolve this problem, adjust your local time (UTC) and try again.</p>
Obs.0005	The server load is too heavy. Try again later.
Obs.0006	<p>The number of buckets has reached the upper limit.</p> <p>An account (including all IAM users under this account) can create a maximum of 100 buckets and parallel file systems. You can use the fine-grained access control of OBS to properly plan and use buckets.</p>
Obs.0007	The target bucket does not exist or is not in the same region with the current bucket.
Obs.0008	The account has not been registered with the system. Only a registered account can be used.
Obs.0009	<p>A conflicting operation is being performed on this resource. Please retry.</p> <p>This is because that there is a bucket with the same name as the bucket you are creating in OBS and the existing bucket has been released in the recent period due to arrears. In such case, try another bucket name.</p>
Obs.0010	Deletion failed. Check whether objects or objects of historical versions exist in the bucket.
Obs.0011	The bucket policy is invalid. Configure it again.
Obs.0012	The requested bucket name already exists. Bucket namespace is shared by all users in the system. Enter a different name and try again.
Obs.0013	The requested folder name already exists. Enter a different name and try again.
Obs.0014	The file size has exceeded 50 MB. Use OBS Browser+ to upload it.
Obs.0015	The absolute path in the search criteria cannot exceed 1023 characters. Please retry.
Obs.0016	<p>Upload failed. Possible causes:</p> <ol style="list-style-type: none"> <li>1. The network is abnormal.</li> <li>2. You have incorrect or no permissions to write the bucket.</li> </ol>



Error Code	Description
Obs.0017	The end time of the new validity period must be later than that of the old validity period.
Obs.0018	The validity period cannot be shorter than the remaining period.
Obs.0019	Cannot determine whether the bucket has objects or fragments. Check whether you have the read permission for this bucket.
Obs.0020	TMS system error. Try again later.
Obs.0021	You do not have permissions to access TMS. Configure the required permissions in IAM.
Obs.0022	The TMS system is busy. Try again later.

# 3 FAQ

---

## 3.1 OBS Basics

### 3.1.1 How Can I Get Started with OBS?

Create an account, add a payment method, and you can start using OBS.

If you use an IAM user, ensure that the user has been added to a user group that has the permissions required to use OBS.

### 3.1.2 How Do I Obtain an OBS Endpoint?

You can access OBS through domain names. When you are using the API, third-party tools, or other methods to access OBS, you can use domain names to conveniently locate resources in OBS.

Before using OBS, ensure that the DNS server address has been correctly configured on the client.

Each data center has its own domain name. For details about domain names, see [Regions and Endpoints](#).

### 3.1.3 What Are the Advantages of Object Storage over SAN and NAS Storage?

- SAN storage provides LUNs or volumes for applications. LUNs and volumes are forms of disk storage. Upper-layer applications use Fibre Channel or iSCSI protocols to access SAN storage. SAN storage focuses on disk management. For other purposes, SAN storage must rely on upper-layer applications.
- NAS storage provides file systems or folders for applications. Upper-layer applications use NFS or CIFS protocols to access NAS storage. Directory trees of file systems must be maintained.
- Object storage is suitable for web applications. A massive bucket storage space is provided based on a URL address to store a wide range of file objects. Object storage adopts a flat architecture. Users do not need to maintain

complex file directories. There is no need to worry about running out of storage because the storage a bucket can provide is practically unlimited.

### 3.1.4 Which Types of Data Can Be Stored in OBS?

OBS can store all types of data.

### 3.1.5 How Much Data Can I Store in OBS?

There are no restrictions on the total capacity or number of objects or files that can be stored by the OBS system or in any single bucket. However, there are limitations on what you can upload to your bucket at a time.

- OBS Console allows you to upload files in a batch. Up to 100 files can be uploaded at a time, with the total size of no more than 5 GB. If you upload only one file in a batch upload, it cannot exceed 5 GB in size.
- If you use OBS Browser+, obsutil, or an API, you can upload a single object of up to 48.8 TB.

### 3.1.6 Does OBS Support Traffic Monitoring?

Yes.

On Cloud Eye, you can monitor the OBS metrics described in the following table.

Metric ID	Metric	Description	Value Range	Monitored Entity	Monitoring Period (Original Metric)
download_bytes	Bytes Downloaded	Specifies the response bytes of all download requests made to all buckets in a region, including bytes in HTTP entity bodies. Unit: byte	≥ 0 bytes	Bucket	5 min
upload_bytes	Bytes Uploaded	Specifies the bytes of all upload requests made to all buckets in a region, including bytes in HTTP entity bodies. Unit: byte	≥ 0 bytes	Bucket	5 min

Metric ID	Metric	Description	Value Range	Monitored Entity	Monitoring Period (Original Metric)
get_request_count	GET Requests	Specifies the number of GET, HEAD, or OPTIONS requests made to all buckets and objects in the buckets of a region. Unit: count	≥ 0 counts	Bucket	5 min
put_request_count	PUT Requests	Specifies the number of PUT, POST, and DELETE requests made to all buckets and objects in the buckets of a region. Unit: count	≥ 0 counts	Bucket	5 min
first_byte_latency	First Byte Download Delay	Specifies the average time from receiving a GET, HEAD, or OPTIONS request to the time that the system starts to respond in a measurement period. Unit: ms	≥ 0 ms	Bucket	5 min
request_count_4xx	4xx Errors	Specifies the times that the server responds to requests whose error codes are 4xx. Unit: count	≥ 0 counts	Bucket	5 min
request_count_5xx	5xx Errors	Specifies the times that the server responds to requests whose error codes are 5xx. Unit: count	≥ 0 counts	Bucket	5 min

### 3.1.7 Can Folders in OBS Be Used the Same Way as in a File System?

No.

OBS does not involve files or folders like in a file system. For your convenience, OBS provides a way to simulate folders. On OBS Console, you can simulate a

folder by adding a slash (/) to the name of an object, which is then displayed as a folder.

### 3.1.8 Where Is Data Stored in OBS?

When creating a bucket on OBS, you can specify a region for the bucket. Then your data on OBS is stored on multiple storage devices in this region.

### 3.1.9 Does OBS Support Access over HTTPS?

Yes, OBS can be accessed over HTTPS.

- When accessing OBS using the allocated domain name, just replace **http** in the URL of the bucket or object with **https** in the browser.

### 3.1.10 Can Other Users Access My Data Stored in OBS?

Yes.

- Bucket ACLs and bucket policies can be used to grant other users read access to your buckets.
- You can grant other users read permissions for objects in your bucket by configuring object ACLs, object policies, or bucket policies. Alternatively, you can configure object sharing.

### 3.1.11 Does OBS Support Resumable Transfer?

Resumable transfer is supported for all transfer methods except API.

**Table 3-1** Support for resumable transfer by different OBS tools

OBS Tool	Resumable Data Transfer
OBS Console	Not supported
OBS Browser+	Supported
obsutil	Supported
APIs	Not supported

### 3.1.12 Does OBS Support Batch Upload?

The following table lists the batch upload support for different OBS tools.

**Table 3-2** Support for batch upload by different OBS tools

Tool	Batch Upload
OBS Console	OBS Console allows you to upload files in a batch. Up to 100 files can be uploaded at a time, with the total size of no more than 5 GB. For details, see <a href="#">Uploading an Object</a> .

Tool	Batch Upload
OBS Browser+	Supports batch upload of files and folders. A maximum of 500 files or folders can be uploaded at a time.
obsutil	Supports upload of a single folder with the maximum size of 48.8 TB.
APIs	Not supported

### 3.1.13 Does OBS Support Batch Download?

The following table lists the batch download support for different OBS tools.

**Table 3-3** Support for batch download by different OBS tools

Tool	Batch Download
OBS Console	Not supported
OBS Browser+	Supported
obsutil	Supported
APIs	Not supported

### 3.1.14 Does OBS Support Batch Deletion of Objects?

The following table lists the batch deletion support for different OBS tools.

**Table 3-4** Support for batch deletion by different OBS tools

Tool	Batch Deletion
OBS Console	Supported. A maximum of 100 objects can be deleted at a time. If a folder is selected, only one folder can be deleted at a time.
OBS Browser+	Supported. Files and folders can be deleted in a batch, and the number of files and folders to be deleted is not limited.
obsutil	You can delete objects in batches by prefix.
APIs	Supported. A maximum of 1,000 objects can be deleted at a time.

 NOTE

The batch deletion performance is negatively correlated with the number of objects in a single request. When it comes to QPS, deleting  $N$  objects is counted as  $N$  operations. If a large number of objects named with prefixes in lexicographic order are deleted, lots of requests may be concentrated in a specific partition, which results in hot access. This limits the request rate in the hot partition and increases access delay.

To address this problem, you can reduce the number of objects in a single batch deletion request, initiate more concurrent requests, and name objects with random prefixes.

### 3.1.15 What Are the Factors That Affect Upload and Download Speed of OBS?

The OBS upload and download speed may be affected by:

- The default upper limit of the OBS read/write bandwidth allowed for a single account: 16 Gbit/s (which means the total GET and PUT bandwidths over both public and private networks)

If the actual bandwidth reaches this upper limit, flow control will be triggered.

- Bandwidth of the purchased VM NIC

If the NIC bandwidth is lower than 16 Gbit/s, the node bandwidth will be limited by the VM bandwidth. You need to purchase multiple VMs to run concurrently to reach 16 Gbit/s.

- Disk I/O and resources consumed by other processes

### 3.1.16 Why Did Some of My Data Stored on OBS Get Lost?

- Check whether there is a lifecycle rule configured to automatically delete objects after a certain date.
- Check whether the write permission to the bucket has been granted to other users. If it was, those other users can delete objects from the bucket. If you have enabled logging, you can check the logs to find out who deleted the objects.

### 3.1.17 Can Deleted Data Be Recovered?

- If versioning is enabled for a bucket, deleted objects are saved to the **Deleted Objects** list. You can recover objects from the **Deleted Objects** list. For details, see [Undeleting an Object](#).
- If versioning is not enabled, deleted objects cannot be recovered.

### 3.1.18 Will There Be Data Left Over in OBS After I Delete an Object?

After you select the objects that you want to delete, OBS will delete the data completely, with nothing remaining. This protects against data leaks.

### 3.1.19 What Are the Differences Between OBS, EVS, and SFS?

[Table 3-5](#) compares OBS, EVS, and SFS.

**Table 3-5** Comparison between OBS, EVS, and SFS

Dimension	OBS	EVS	SFS
Concept	OBS provides massive, secure, reliable, and cost-effective data storage for users to store data of any type and size.	EVS provides scalable block storage that features high reliability, high performance, and robust specifications for ECSs to meet service requirements in different scenarios. An EVS disk is similar to a hard disk on a PC.	SFS provides on-demand high-performance file storage, which can be shared by multiple ECSs. SFS is similar to a remote directory for a Windows or Linux machine.
Data storage logic	Stores objects. Files can be stored directly to OBS. The files automatically generate corresponding system metadata. You can also customize the metadata if needed.	Stores binary data and cannot store files directly. To store files on an EVS disk, you need to format the file system first.	Stores files. Data is sorted and displayed in files and folders.
Access mode	You can access OBS over the Internet or using Direct Connect. Just specify the bucket address and use a transmission protocol, for example, HTTP or HTTPS.	EVS disks need to be attached to an ECS or BMS and initialized before they can be used and accessed by your applications.	SFS systems need to be mounted to an ECS or BMS and then they can be accessed using NFS or CIFS protocols. A network address must be specified or mapped to a local directory for access.



Dimension	OBS	EVS	SFS
Application scenario	Big data analysis, static website hosting, online video on demand (VoD), gene sequencing, and intelligent video surveillance	HPC, enterprise core cluster applications, enterprise application systems, and development and testing  <b>NOTE</b> HPC: High-speed and high-IOPS storage is required, such as industrial design and energy exploration.	High-performance computing (HPC), media processing, file sharing, content management, and web services  <b>NOTE</b> HPC: High bandwidth is required for shared file storage, such as gene sequencing and image rendering.
Capacity	Exabytes	Terabytes	Petabytes
Latency	Milliseconds	1 to 2 ms	3 to 10 ms
IOPS/TPS	10 million	50,000 for a single disk	10,000 for a single file system
Bandwidth	TB/s	MB/s	GB/s
Data sharing supported	Yes	Yes	Yes
Remote access supported	Yes	No	Yes
Online editing supported	No	Yes	Yes
Used independently	Yes	No	Yes

### 3.1.20 Will My Bucket Performance Be Affected by Other Users' Services?

No. OBS isolates the access from different accounts, so there is no performance interference or impact between different accounts.

## 3.2 Access Control

### 3.2.1 How Can I Control Access to OBS?

You can use the following mechanisms to control access to OBS.

- IAM policies  
IAM policies define the actions that can be performed on your cloud resources, specifying what actions are allowed or denied.  
IAM policies can be used to grant access to various IAM users under the same parent account.  
The process is as follows:
  - a. Create a user group and select an IAM permission set for it.
  - b. Create an IAM user and add it to the user group, and it will inherit the permissions of the user group you added it to.
- Bucket policies  
A bucket policy applies to the configured OBS bucket and all the objects in the bucket. An OBS bucket owner can use a bucket policy to grant permissions on buckets and objects in the buckets to IAM users or other accounts.
- Access Control List (ACL)  
ACLs control read and write permissions for accounts. ACL control is not as fine-grained as bucket policies and IAM policies, so IAM policies and bucket policies are recommended instead.

### 3.2.2 What Are the Differences Between Using an IAM Policy and a Bucket Policy in Access Control?

IAM policies apply to cloud resources. With the OBS permissions, an IAM policy can be applied to all buckets and objects, or it can be applied only to specified buckets and objects. IAM policies are recommended if you assign permissions to IAM users of the same account.

A bucket policy only applies to the bucket the policy was configured for.

### 3.2.3 What Is the Relationship Between a Bucket Policy and an Object Policy?

An object policy takes effect on only one object in a bucket. A bucket policy can be applied to multiple or all objects in a bucket.

### 3.2.4 Why Is the Message "Access denied" Still Appearing After OBS System Permissions Were Assigned by IAM?

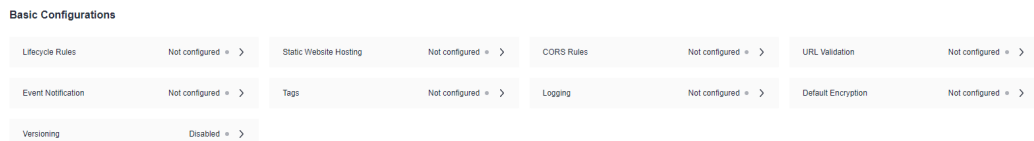
#### Cause

System permissions such as OBS ReadOnlyAccess, OBS OperateAccess, and OBS Buckets Viewer preset in IAM only allow certain OBS operations. For example, the OBS OperateAccess permission lets you list buckets, obtain basic bucket information, obtain bucket metadata, list objects (not the objects that have been versioned), upload objects, download objects, delete objects, and obtain object ACLs. Performing each operation requires calling an OBS API.

After your account has been granted system permissions, you can call these APIs directly or through SDKs. However, when you log in to OBS Console or use OBS

Browser+, more APIs are called to load the bucket list or the bucket's overview page. If your permissions do not cover those APIs, your access is denied, or you receive a message indicating that the operation is not allowed. For example, loading the bucket's overview page involves API calls to query the configuration statuses of lifecycle and CORS rules. See [Figure 3-1](#). However, the preset system permissions do not cover these operations.

**Figure 3-1** Basic bucket configurations



Basic Configurations			
Lifecycle Rules	Not configured >	Static Website Hosting	Not configured >
Event Notification	Not configured >	Tags	Not configured >
Versioning	Disabled >	CORS Rules	Not configured >
		Logging	Not configured >
		URL Validation	Not configured >
		Default Encryption	Not configured >

## Solutions

Authorized permissions are valid, though operations on the console or client are restricted. You can call the APIs directly or through SDKs.

On OBS Console or OBS Browser+ (a client), the OBS OperateAccess permission allows you to upload and download objects.

If you do not want those error messages to appear, you can configure policies by referring to [Configuring Fine-Grained Policies](#) on the IAM console to grant more OBS permissions to a user group, and add the user who requires the permissions to this group.

## Why Can't I List Objects on OBS Console Even If I Have Been Granted the OBS OperateAccess and OBS ReadOnlyAccess Permissions?

System policies OBS OperateAccess and OBS ReadOnlyAccess contain only `obs:bucket:ListBucket` (used to list objects), but do not contain `obs:bucket:ListBucketVersions` (used to list multiple versions of objects).

If a bucket has multiple versions of objects, IAM users may fail to list objects in the bucket through OBS Console. In such case, IAM users need to be granted the `obs:bucket:ListBucketVersions` permission.

## 3.2.5 Why Does Message "Access denied" Appear After I Was Granted the Read and Write Permissions for a Bucket?

### Cause

If you use a bucket policy to grant the IAM user the bucket read and write permissions, the IAM user has the permissions to call the following APIs:

- `GetObject`: downloading objects
- `GetObjectVersion`: downloading objects and their versions
- `PutObject`: uploading objects
- `DeleteObject`: deleting objects
- `DeleteObjectVersion`: deleting objects and their versions

Each API requires an operation permission. IAM users can call these APIs directly or through SDKs. However, when you log in to OBS Console or using a client tool such as OBS Browser+, more APIs (such as ListAllMyBuckets and ListBucket) are called to load the bucket list and object list. If your permissions do not cover those APIs, your access is denied, or you are informed that the operation is not allowed.

## Solutions

Authorized permissions are valid, though operations on the console or client are restricted. You can call the APIs directly or through SDKs.

If you want to access OBS through OBS Console or OBS Browser+ (a client), you can configure policies by referring to [Configuring Fine-Grained Policies](#) on the IAM console to grant more OBS permissions to a user group, and add the user who requires the permissions to this group.

### 3.2.6 Why Can't I Access OBS (403 AccessDenied) After Being Granted with the OBS Access Permission?

#### Problem Description

By configuring IAM permissions, bucket policies, or bucket ACLs, you have been granted the permissions needed to access OBS. However, when you try to access OBS, the error message **Access denied** or **403 AccessDenied** is displayed.

#### Problem Analysis

Possible causes are described here in order of how likely they are to occur. To locate the root cause as fast as possible, go through the list in order, from most likely to least.

If the fault persists after a possible cause is rectified, move down the list to the next most likely cause.

#### NOTE

**Table 3-6** Problem Analysis

Possible Cause	Solution
The permissions did not take effect due to IAM caching.	Due to data caching, it can take about 10 to 15 minutes for a new IAM permission configuration to take effect. Try again in 10 to 15 minutes.

Possible Cause	Solution
An incorrect account or access key (AK or SK) was used to access OBS.	If you do not have the permissions needed to access OBS, the login information, such as the account or AK/SK used was likely incorrect. Incorrect use of AK/SK is more common. For example, you may be using an AK/SK or password for a different account.  Confirm the login credentials with the resource owner.
The permissions were incorrectly configured.	For details, see <a href="#">Checking Whether Permissions Are Correctly Configured</a> .
URL validation was configured.	Modify the Referer field in the whitelist or blacklist by referring to <a href="#">Configuring URL Validation</a> .

## Checking Whether Permissions Are Correctly Configured

OBS provides multiple mechanisms for permissions management, and in some scenarios there may be dependencies involved. If you cannot access OBS, contact the person who assigned the permissions (usually the resource owner) to check whether the permissions were configured correctly. There are two critical elements to check: **Resources** (which resources access is granted to) and **Actions** (authorized operations). For commonly seen mistakes, see [Table 3-7](#). If **Condition** is configured in the IAM permission or bucket policy, check whether the specified rules are met.

**Table 3-7** Commonly seen mistakes in configuring **Resources** and **Actions**

Type	Common Mistake
Resources	<ul style="list-style-type: none"><li>You were granted access to a given bucket but tried to access a different bucket.</li><li>You were granted access to a bucket but not to the objects in that bucket.</li><li>You were granted access to view a bucket but not to perform any operations (for example, listing objects in the bucket).</li><li>You were only granted access to certain objects in a bucket but tried to access other objects.</li></ul>

Type	Common Mistake
Actions	<ul style="list-style-type: none"><li>• Actions configured incorrectly: For example, the download permission (GetObject) may have been mistakenly assigned instead of the upload permission (PutObject).</li><li>• Required actions are missing from the configuration: Some permissions required by OBS Console and OBS Browser+ for other actions are often ignored. The most often ignored actions are <b>ListAllMyBuckets</b> and <b>ListBucket</b>, which are needed for viewing a list of the buckets and of the objects in those buckets. Typical examples are described in:<ul style="list-style-type: none"><li>– <a href="#">Why Is the Message "Access denied" Still Appearing After OBS System Permissions Were Assigned by IAM?</a></li><li>– <a href="#">Why Does Message "Access denied" Appear After I Was Granted the Read and Write Permissions for a Bucket?</a></li></ul></li></ul>

### Checking IAM permissions

1. On the top menu bar, choose **Service List > Management & Deployment > Identity and Access Management**. The IAM console is displayed.
2. On the **Users** page, search for the name of the user that could not access OBS. Click the name to check which user group the user belongs to.
3. On the **User Groups** page, search for the user group to which the user belongs. In the **Operation** column of the user group, click **Manage Permissions** to see which IAM permissions have been granted.

### Checking the bucket policy

1. In the service list, choose **Storage > Object Storage Service**.
2. In the bucket list, search for the bucket that fails to be accessed and click the bucket name. The **Objects** page is displayed.
3. In the navigation pane, choose **Permissions > Bucket Policies** to view the configured bucket policies.

### Checking the bucket ACL

1. In the service list, choose **Storage > Object Storage Service**.
2. In the bucket list, search for the bucket that fails to be accessed and click the bucket name. The **Objects** page is displayed.
3. In the navigation pane, click **Permissions**. The **Bucket Policies** tab page is displayed. Then go to the **Bucket ACLs** page to view the configured bucket ACLs.

### Checking the object ACL

1. In the service list, choose **Storage > Object Storage Service**.

2. In the bucket list, search for the bucket that fails to be accessed and click the bucket name. The **Objects** page is displayed.
3. In the object list, search for the object that fails to be accessed and click the object name. On the page that is displayed, view the object ACL configuration on the **Object ACL** tab.

### 3.2.7 How Do I Control Access to Folders in an OBS Bucket?

You can customize a bucket policy and specify a prefix in it to control access to folders.

For example, if the prefix is set to **abc/**, configured permissions will apply to folder **abc**.

For details, see [Creating a Custom Bucket Policy \(Visual Editor\)](#).

## 3.3 Buckets and Objects

### 3.3.1 Why Am I Unable to Create a Bucket?

- If the number of buckets created by the current user reaches 100, delete some unneeded buckets first.
- If the name for the new bucket already exists, use another name and try again. Each OBS bucket name must be globally unique. Specifically, it must be different from that of buckets created by its owner or by any other users.
- The name of a deleted bucket cannot be reused immediately after the deletion. It can be reused for a bucket or a parallel file system at least 30 minutes later after the deletion.
- Check whether the account has required permissions. If the account does not have the permissions, grant them.
- Check whether the account is in arrears or the account balance is insufficient. If this is the case, pay off the outstanding balance or top up the account.
- Check whether the network connectivity between the local computer and OBS is normal. If the network is down, restore the network connection.
- If the failure is not caused by any of the preceding reasons, check the returned error code and find the reason.

### 3.3.2 Why Am I Unable to Upload an Object?

- Check whether the network connectivity between the local computer and OBS is normal. If the network is down, restore the network connection.
- If a message indicating "service unavailable" is displayed when objects are being uploaded, try again later.
- Check whether the account is in arrears or the account balance is insufficient. If this is the case, pay off the outstanding balance or top up the account.
- Check whether the account has the permissions required to upload objects. This check should cover the IAM policies, bucket policies, and bucket ACLs. If the account does not have the required permissions, grant the permissions first.

- If the fault persists, contact customer service.

### 3.3.3 Why Am I Unable to Download an Object?

- Check whether the network connectivity between the local computer and OBS is normal. If the network is down, restore the network connection.
- Check whether the account is in arrears or the account balance is insufficient. If this is the case, pay off the outstanding balance or top up the account.
- Check whether the account has the permissions needed to download objects from the bucket. This check should cover IAM policies, bucket policies, object policies, bucket ACLs, and object ACLs. If the account does not have the required permissions, grant the permissions first.
- Check whether the object is in the Cold storage class. If it is and the status is **Unrestored**, restore the object first.
- If the fault persists, contact customer service.

### 3.3.4 Why Can't I Delete a Bucket?

- Check whether the network connectivity between the local computer and OBS is normal. If the network is down, restore the network connection.
- Check whether all objects in the bucket have been deleted. If not, delete all objects from the bucket.
- Check whether all fragments in the bucket have been deleted. If not, delete all fragments from the bucket.
- If versioning is enabled, check whether there are deleted objects remaining in the bucket. If yes, permanently delete all deleted objects from the bucket.
- Check whether the account that deletes the bucket is the owner of the bucket.
- If the fault persists, contact customer service.

### 3.3.5 What Is the Relationship Between Bucket Storage Classes and Object Storage Classes?

When an object is uploaded, it inherits the storage class of the bucket by default, but you can change the default storage class when you upload the object.

Changing the storage class of a bucket does not change the storage classes of existing objects in the bucket, but newly uploaded objects will inherit the new storage class.

### 3.3.6 Can I Modify the Region of a Bucket?

No. After a bucket is created, the region cannot be changed.

### 3.3.7 Can I Edit Objects in OBS Online?

OBS is a cloud storage service. It provides massive, secure, highly reliable, and low-cost data storage capabilities.



Generally, OBS does not support online editing of object content. You can download the object that you want to edit to a local path, modify the object, and then upload it to OBS again.

There are a few exceptions. Online, you can use OBS to:

1. **Modify object metadata**, such as **ContentDisposition** and **ContentLanguage**.
2. Process images stored in OBS.
3. Add data to the end of an object. For details, see **Appending an Object** in the *Object Storage Service API Reference*.

### 3.3.8 How Do I Obtain the Access Path to an Object?

Object access paths use the following format: **https://{bucket name}.{domain name}/{object name}**.

You can combine a path manually or use the tools in the following table to obtain it.

**Table 3-8** How to obtain an object URL

Tool	Object URL
OBS Console	Click the object and copy the URL for the detailed information of the object.
OBS Browser+	Click the <b>Attribute</b> button of the object and then you can copy the URL displayed in the detailed information about the object.
obsutil	Not supported
APIs	Not supported

#### NOTE

If the object access path is user-assembled, you need to escape the object name by referring to the URL encoding rules.

### 3.3.9 Why Can't I Search for Certain Objects in My Bucket?

On OBS Console and OBS Browser+, you can search for objects by object name prefix. For example, if you search for **test**, you will find all objects whose names start with **test**. However, if the keyword entered is in the middle or at the end of the object name, the search will not return those results. For example, the name of the object to be searched for is **testabc** and you enter **abc** in the search box, **testabc** will not be found. Only objects whose names start with the prefix **abc** are found.

### 3.3.10 What Should I Do If an Error Message Is Displayed When I Use Internet Explorer to Access an Object URL That Contains Chinese Characters?

#### Description

HTTP 400 error is returned when using the Internet Explorer to access an object URL that contains Chinese characters?

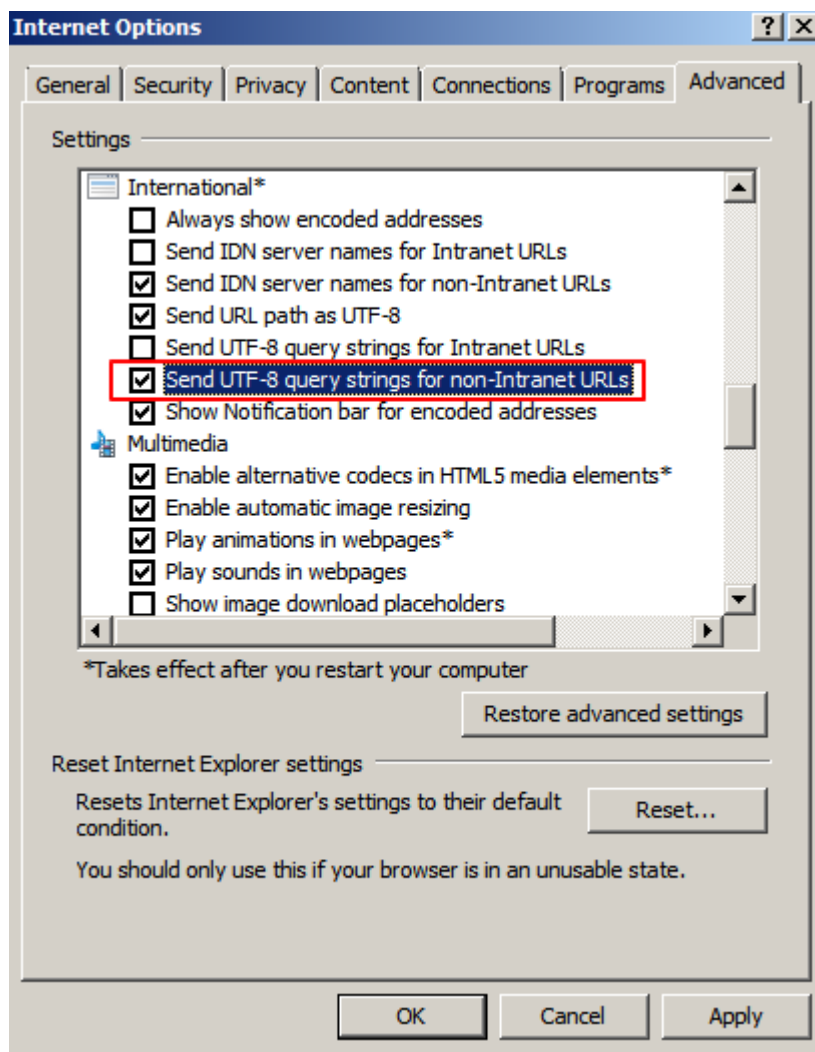
#### Handling Method

By default, the Internet Explorer does not use the UTF-8 to send query strings. To solve this problem, change the default configuration of the Internet Explorer.

#### Procedure

- Step 1** Open Internet Explorer, for example, IE 11.
- Step 2** Click **Settings** in the upper right corner of the browser and choose **Internet Options > Advanced**.
- Step 3** Select **Send UTF-8 query string for non-Internet URLs**, as shown in the following figure.

Figure 3-2 Changing IE default settings



**Step 4** Click **Apply**, and then click **OK**.

**Step 5** Restart Internet Explorer.

Then, you can properly access the object URL.

----End

### 3.3.11 How Do I Batch Delete a Large Number of Objects from a Bucket or Empty a Bucket?

You can batch delete a large number of objects from a bucket or empty a bucket by referring to the procedure below:

#### Method: Using Lifecycle Rules

You can use the OBS [lifecycle management](#) to periodically empty all objects in a bucket at a time or batch delete objects based on a specified prefix.

**Step 1** In the navigation pane of OBS Console, choose **Object Storage**.

- Step 2** In the bucket list, click the bucket you want to operate to go to the **Objects** page.
- Step 3** In the navigation pane, choose **Overview**.
- Step 4** In the **Basic Configurations** area, click **Lifecycle Rules**. The **Lifecycle Rules** page is displayed.
- Step 5** Click **Create**. A dialog box shown in **Figure 3-3** is displayed.

**Figure 3-3** Creating a lifecycle rule

- Step 6** Configure a lifecycle rule for emptying a bucket or batch deleting objects with a specified prefix.

**Table 3-9** Lifecycle rule parameters

Category	Parameter	Description
Basic information	Status	Select <b>Enable</b> .
	Rule Name	User-defined. It identifies a lifecycle rule.
	Prefix	Optional. <ul style="list-style-type: none"> <li>● If this parameter is configured, objects with the specified prefix will be deleted in a batch.</li> <li>● If this parameter is not configured, all objects in the bucket will be deleted.</li> </ul>

Category	Parameter	Description
Current version/ Historical version	Transition to Warm After (Days)	Do not select this parameter.
	Transition to Cold After (Days)	Do not select this parameter.
	Delete Objects After (Days)	Select this parameter and specify a number. It indicates the number of days after the last update when objects are automatically deleted. The minimum value is <b>1</b> day. If any of the transition operations is configured, this parameter must be set to a number larger than that specified for any of the transition operations.  <b>NOTE</b> If versioning is not enabled for the current bucket, specified objects will be automatically deleted after they expire and cannot be recovered.

 **NOTE**

- **Current version** and **Historical version** are two concepts for **Versioning**. If **Versioning** is enabled, uploading objects with the same name to the same path generates different versions of the same object. The object uploaded lastly is called **Current Version**, and the object uploaded previously is called **Historical Version**.
- The **Historical Version** appears only when the versioning is enabled or suspended for the bucket.
- Either the **Current Version** or **Historical Version** must be configured, or you can configure both of them. If you want to empty the bucket, configure both of them.
- There may be a delay for deleting objects after the objects expire. The delay generally does not exceed 48 hours. If you change the configurations of an existing lifecycle rule, the rule will take effect again.

**Step 7** Click **OK** to complete the configuration.

----End

## 3.4 Tools

### 3.4.1 When Downloading a Folder Using obsutil, the Download Speed Slows After the Folder Download Progress Reaches 90%

This problem may occur in the following scenarios:

- Scenario 1: The folder contains a few large objects among a large number of small objects. Large objects are downloaded at fast speed. But the download

speed of small objects in large quantity is closely related to the TPS performance. Therefore, if the remaining 10% are mostly small objects, the download speed may decrease.

- Scenario 2: The folder contains same-size objects. It is possible that all objects have been downloaded but are queuing to be written to disks, which may be reflected as a slowdown in the download progressing. In this case, check the writing speed of your clients.

### 3.4.2 With `obsutil`, Downloading a File Fails After the Download Progress Reaches 99%

#### Possible causes:

1. Network fluctuation
2. Failure in caching the file to the target folder due to disk I/O freezes.

#### Solution:

1. Run the download command again.  
The resumable download function is enabled by default for `obsutil` download tasks. You only need to run the same download command again, the failed file download will be resumed and the file will be downloaded to your local path.
2. If the problem persists, upgrade `obsutil` to the latest version and try again.

### 3.4.3 How Do I Use the `obsutil cp` Command to Enable Incremental Upload, Download, or Replication?

When running the `obsutil cp` command to upload or download data, you can add the `-u` parameter to enable the incremental upload/download function.

This parameter indicates that the system will compare the source path with the target path when uploading, downloading, or replicating an object. The system uploads, downloads, or replicates an object only when the target object does not exist, the object size is inconsistent, or the last modification time of the target object is earlier than that of the source object.

## 3.5 APIs and SDKs

### 3.5.1 What Are the Differences Between PUT and POST Upload Methods?

Parameters are passed through the request header if the PUT method is used to upload objects; if the POST method is used to upload objects, parameters are passed through the form field in the message body.

With the PUT method, you need to specify the object name in the URL, but object name is not required with the POST method, which uses the bucket domain name as the URL. Request lines of these two methods are given as follows:

```
PUT /ObjectName HTTP/1.1
```

POST / HTTP/1.1

Either PUT or POST method allows the object size of [0, 5 GB] for each upload. If you need to upload an object greater than 5 GB, use the multipart upload method.

For details about PUT and POST APIs, see the *Object Storage Service API Reference*.

## 3.5.2 Failure with OBS SDK in Uploading a File Greater than 5 GB

OBS server has a restriction on the object upload API, which only allows a maximum of 5 GB for an upload. If you want to upload a file greater than 5 GB, use the multipart upload API. Operations are detailed in the following procedure:

1. Call the OBS API for initializing a multipart upload task to generate a multipart upload ID (Upload ID).
2. Call the OBS API for uploading parts one by one or in parallel. The size of each part can be up to 5 GB.
3. After parts are uploaded, call the OBS API to merge parts to generate the complete object.

OBS SDKs support atomic operations. In the section "Multipart Upload" of *OBS SDK Reference* in different programming languages, you can find more information about how to implement multipart upload using OBS SDKs.

## 3.5.3 Why Don't the Signatures Match?

### Symptom

The following error is reported during an OBS API call.

Status code: 403 Forbidden

Error code: SignatureDoesNotMatch

Error message: The request signature we calculated does not match the signature you provided. Check your key and signing method.

### Possible Causes

The provided signature does not match the signature calculated by the system.

### Solution

**Step 1** Check the endpoint.

Check the endpoint if you are using the OBS SDK.

Ensure that the entered **endpoint** is correct. If the endpoint is set to a bucket domain name that consists of a bucket name and an endpoint, a signature mismatch error will also be reported.

**Step 2** Check the AK and SK.

Ensure that the AK and SK you entered are correct, so they can match those used in the request.

**Step 3** Check **HTTP-Verb**.

Ensure that the **HTTP-Verb** in the signature is the same as that in the request.

**Step 4** Check **Date** and **Expires**.

- Signature in a header: Check whether the **Date** in the signature is the same as that in the request header.
- Signature in a URL: Check whether the **Expires** in the signature is the same as that in the request URL.

**Step 5** Check headers.

Check **Content-MD5**, **Content-Type**, and **Canonicalized Headers**. If any of them are contained during signature calculation, they must be also contained in the request.

 **NOTE**

If a URL with a signature contained is used to access OBS resources through a browser, the header parameters above cannot be contained during signature calculation.

**Step 6** Check **Canonicalized Resource**.

**Canonicalized Resource** indicates the OBS resources that are requested. Configure this parameter based on the requirements in the API reference. For details, see section "Authentication of Signature in a Header" or "Authentication of Signature in a URL" in the *Object Storage Service API Reference*.

**Step 7** Check **StringToSign**.

Check whether **StringToSign** is constructed based on the following rules:

- Signature in a header:  
HTTP-Verb + "\n" + Content-MD5 + "\n" + Content-Type + "\n" + Date + "\n" + CanonicalizedHeaders + CanonicalizedResource
- Signature in a URL:  
HTTP-Verb + "\n" + Content-MD5 + "\n" + Content-Type + "\n" + Expires + "\n" + CanonicalizedHeaders + CanonicalizedResource

 **NOTE**

If a parameter is left blank, put it in a new line.

**Step 8** Check the signature calculation.

Check whether the signature is calculated as follows:

1. Construct the request string **StringToSign**.
2. Perform UTF-8 encoding on the result in the **1**.
3. Use the SK to perform the HMAC-SHA1 signature calculation on the result in **2**.
4. Perform Base64 encoding on the result in **3**. If the signature is contained in a header, this step generates the final signature and no further actions are required.



5. If the signature is contained in a URL, perform the URL encoding on the result in [4](#) to obtain the final signature.

----End

## 3.6 Security

### 3.6.1 How Is Data Security Ensured in OBS?

OBS is secure. It provides end-to-end security services. For example, if a bucket or an object is undisclosed when you access the bucket or object, only the owner of the bucket or object can access it. Further, the access to the bucket or object requires access keys (AK/SK). You can also use various access control mechanisms (such as bucket policies and ACLs) to select users and user groups and grant them permissions. OBS supports data transfer over the HTTPS/SSL protocol. Data encryption prior to upload is available to meet your higher security requirements.

### 3.6.2 Does OBS Scan My Data for Other Purposes?

OBS only determines whether data blocks exist or are damaged (repairs data if damaged) by scanning for the data. It does not read specific data.

### 3.6.3 Can Engineers Export My Data from the Background of OBS?

No. Background engineers cannot export your data. For example, if a bucket or an object is undisclosed when you access the bucket or object, only the owner of the bucket or object can access it. Further, the access to the bucket or object requires access keys (AK/SK).

### 3.6.4 How Does OBS Protect My Data from Being Stolen?

Only the owner of a bucket or an object can access it. Accessing a bucket or object requires access keys (AK/SK). In addition, multiple access control mechanisms such as the ACLs, bucket policies, and URL validation are used to ensure data access security.

### 3.6.5 Can a Pair of AK and SK Be Replaced When It Is Being Used to Access OBS?

Yes. The pair of AK and SK can be replaced at any time.

### 3.6.6 Can Multiple Users Share One Pair of AK and SK to Access OBS?

Yes. Different users can use the same pair of AK and SK to access the same resources in OBS.

## 3.7 Durability and Availability

## 3.7.1 What Are the Differences Between Single-AZ and Multi-AZ Storage in OBS?

### Question 1:

Q: For selecting data redundancy policy upon bucket creation, what is the difference between single-AZ storage and multi-AZ storage?

A: Multi-AZ storage means data is stored in multiple AZs, improving data reliability. Single-AZ storage means data is stored in a single AZ, with lower costs.

### Question 2:

Q: Is data stored as copies in multiple AZs when the data redundancy policy is set to multi-AZ storage? If an AZ is faulty, is data complete in other AZs?

A: The Erasure Code (EC) algorithm, instead of multiple copies, is used to ensure data redundancy in the multi-AZ mode. If the multi-AZ storage is enabled for a bucket, data is stored in multiple AZs in the same region. If an AZ is unavailable, data can still be properly accessed in other AZs. The multi-AZ mode is suitable for data storage scenarios that require high reliability. The multi-AZ storage tolerates only faults of a single AZ.

### Question 3:

Q: Can I change the data redundancy policy without deleting the bucket?

A: No. Once a bucket is created, you cannot change the data redundancy policy. You can create a bucket with the wanted data redundancy policy, and migrate data to the new bucket.

## 3.7.2 What Redundancy Storage Techniques Does OBS Use?

OBS uses the Erasure Code (EC) algorithm, instead of multiple copies, to ensure data redundancy.

Compared with the multi-copy redundancy, EC delivers a higher storage space utilization while maintaining the same reliability level.

A bucket with single-AZ storage uses the EC algorithm for data redundancy among nodes in one AZ. A bucket with multi-AZ storage not only ensures redundancy for the data among nodes in an AZ, but also across multiple AZs.

## 3.8 How Do I Use Fragment Management?

### 3.8.1 Why Are Fragments Generated?

Fragments are incomplete data in buckets generated due to data upload failures.

Data can be uploaded to OBS using multipart uploads. There will be fragments generated, if a multipart upload fails because of the following causes (included but not limited to):

- The network is in poor conditions, and the connection to the OBS server is interrupted frequently.

- The upload task is manually suspended.
- The device is faulty.
- The device is powered off suddenly.

## 3.8.2 How Do I Manage Fragments?

Generated fragments take up storage space that is billable.

You can clear the fragments in a bucket on OBS Console or OBS Browser+.

If fragments are generated due to interruptions of multipart upload tasks on OBS Browser+, they will disappear once those tasks are continued and finished.

## 3.9 How Do I Use Versioning?

### 3.9.1 Can I Upload an Object to a Folder Where a Namesake Object Already Exists?

If versioning is enabled and an object is being uploaded, OBS automatically allocates a unique version ID to the object. Objects with the same name are stored in OBS with different version IDs.

If versioning is not enabled and objects with the same name are being uploaded to a specific folder, the new object will overwrite the existing one.

### 3.9.2 Can I Recover a Deleted Object?

When versioning is enabled, if you delete an object without specifying a version ID, the object is tagged with a delete marker and displayed in the list of **Deleted Objects**. You can recover the object from that list.

If you delete an object with a version ID specified when versioning is enabled or you delete an object when versioning is not enabled, OBS permanently deletes the object, and you cannot recover it.

For details, see [Versioning Overview](#).

## 3.10 How Do I Use Tags?

### 3.10.1 Can I Search for a Bucket by Tag?

No. This function is not supported yet.

### 3.10.2 What Can I Do with Tags?

If you add tags to a bucket, service detail records (SDRs) generated for it will be labeled with these tags. You can classify SDRs by tag for cost analysis. For example, if you have an application that uploads its running data to a bucket, you can tag the bucket with the application name. In this manner, the costs on the application can be analyzed using tags in SDRs.

## 3.11 Event Notification

### 3.11.1 Which Events Can Trigger Event Notifications?

OBS supports notification for the following event types:

- **ObjectCreated:** Indicates all kinds of object creation operations, including PUT, POST, and COPY of objects, as well as the merging of parts.
  - **Put:** Creates or overwrites an object using the PUT method.
  - **Post:** Creates or overwrites an object using the POST (browser-based upload) method.
  - **Copy:** Creates or overwrites an object using the COPY method.
  - **CompleteMultipartUpload:** Merges parts of a multipart upload.
- **ObjectRemoved:** Deletes an object.
  - **Delete:** Deletes an object with a specified version ID.
  - **DeleteMarkerCreated:** Deletes an object without specifying a version ID.

For details about how to configure event notifications, see [Configuring Event Notifications](#).

## 3.12 How Do I Use Lifecycle Management?

### 3.12.1 What Are the Application Scenarios of Lifecycle Management?

You may configure lifecycle rules to:

- Periodically delete logs that are only meant to be retained for a specific period of time (a week or a month).
- Transition documents that are seldom accessed to the Warm or Cold storage class or delete them.

If you want to delete a large number of objects from a bucket, you can configure a lifecycle rule to automatically delete the expired objects. [Table 3-10](#) lists the parameters for configuring such a lifecycle rule on OBS Console.

**Table 3-10** Parameters for deletion upon expiration

Parameter	Value
Status	Enable
Rule Name	Example: <b>rule-delete</b>

Parameter		Value
Prefix		Optional. <ul style="list-style-type: none"> <li>If this parameter is configured, objects with the specified prefix will be deleted in a batch.</li> <li>If this parameter is not configured, all objects in the bucket will be deleted.</li> </ul>
Current Version	Delete Objects After (Days)	1 day
Historical Version	Delete Objects After (Days)	1 day

One day later, objects in the bucket are successfully deleted based on the rule. If you do not need this lifecycle rule, you can disable it or delete it.

## 3.13 How Do I Use Static Website Hosting?

### 3.13.1 Can OBS Host My Static Websites?

OBS supports static website hosting. You can configure the static website hosting function for your buckets on OBS Console. When a client accesses objects from the website address of a bucket, the browser can directly resolve the web resources and present them to end users.

### 3.13.2 Which Types of Websites Can I Use OBS to Host?

Static websites contain static web pages and some scripts that can run on clients, such as JavaScript and Flash.

### 3.13.3 How Do I Obtain the Static Website Hosting Address of a Bucket?

You can obtain the static website hosting address of the bucket on OBS Console.

You can also get the address according to the following rule and format. Address format: `https://Bucket name.Domain name of the hosted static website`

## 3.14 How Do I Manage Domain Names?

### 3.14.1 Why Is the Message "NoSuchBucket" Displayed When I Use a User-Defined Domain Name to Access a Bucket That Can Be Accessed by the OBS Domain Name?

The CNAME resolution is not configured, after the domain name is bound to your OBS bucket.

### 3.14.2 What Is the Relationship Between OBS Bucket Names and Domain Names?

An OBS bucket name is the name of the bucket you created.

The domain name is the endpoint of the region where the bucket is located.

The domain name of your bucket is the bucket name plus the regional domain name (*bucket\_name.domain\_name*).

## 3.15 Monitoring

### 3.15.1 Why Can't I Find the Statistics on OBS 5XX Status Codes on Cloud Eye?

OBS metrics on Cloud Eye are displayed based on your requests. Once you perform a request or storage action, Cloud Eye will display the corresponding request or storage metric. For instance, if the server returns a 5XX status code to you, the metric for measuring the number of 5XX status codes will appear on Cloud Eye.

## 3.16 Server-Side Encryption

### 3.16.1 Does OBS Support Encrypted Upload?

OBS provides server-side encryption function. You can encrypt objects while uploading. Data is encrypted on the server and then stored in OBS. When downloading the encrypted objects, the encrypted data will be decrypted on the server and displayed for you in plaintext.

**Table 3-11** lists the encryption methods supported by OBS Console, clients, and tools.

**Table 3-11** Object upload encryption in different access modes

Access Mode	Support for Upload Encryption	Reference
OBS Console	Yes	<a href="#">Uploading an Object in Server-Side Encryption Mode</a>

Access Mode	Support for Upload Encryption	Reference
OBS Browser+	No Object encryption is not supported for upload. However, if the default encryption function is enabled for a bucket, objects uploaded to the bucket will be automatically encrypted.	-
obsutil	No Object encryption is not supported for upload. However, if the default encryption function is enabled for a bucket, objects uploaded to the bucket will be automatically encrypted.	-
API	Yes	See section "API Operations Related to Server-Side Encryption" in the <i>Object Storage Service API Reference</i> .

### 3.16.2 What Encryption Technologies Can I Use to Encrypt Data on OBS?

Before uploading your data to OBS, you can encrypt the data to ensure security during transmission and storage. OBS support various encryption technologies used on clients.

OBS allows you to encrypt objects with server-side encryption so that the objects can be securely stored in OBS.

The objects to be uploaded can be encrypted using SSE-KMS. You need to create a key using KMS or use the default key provided by KMS. Then you can use the KMS key to perform server-side encryption when uploading objects to OBS.

After server-side encryption is enabled, objects to be uploaded will be encrypted and stored on the server. When objects are downloaded, they will be decrypted on the server first and then returned in plaintext to you.

OBS provides SSE-KMS and SSE-C that can be configured by calling APIs. With SSE-C, OBS uses the customer-provided keys and their MD5 values for server-side encryption.

### **3.16.3 Will OBS Server-Side Encryption Encrypt My Existing Objects That Are Unencrypted?**

No. OBS encrypts only objects that are uploaded after the server-side encryption configuration is effective. If you want to encrypt existing objects, delete them and upload them again.



# A Change History

Release Date	What's New
2024-01-29	<p>This issue is the eighth official release.</p> <p>This issue incorporates the following changes:</p> <ul style="list-style-type: none"><li>• Adapted to the new OBS Console.</li><li>• Added the description of image processing.</li><li>• Added the introduction to tag functions in <a href="#">Tags</a>.</li><li>• Added the description of Task Center in <a href="#">Task Center</a>.</li><li>• Added the following FAQs:<ul style="list-style-type: none"><li>- <a href="#">Will My Bucket Performance Be Affected by Other Users' Services?</a></li><li>- <a href="#">Why Is the Message "Access denied" Still Appearing After OBS System Permissions Were Assigned by IAM?</a></li><li>- <a href="#">Why Does Message "Access denied" Appear After I Was Granted the Read and Write Permissions for a Bucket?</a></li><li>- <a href="#">Why Can't I Access OBS (403 AccessDenied) After Being Granted with the OBS Access Permission?</a></li><li>- <a href="#">How Do I Control Access to Folders in an OBS Bucket?</a></li><li>- <a href="#">Can I Edit Objects in OBS Online?</a></li><li>- <a href="#">How Do I Batch Delete a Large Number of Objects from a Bucket or Empty a Bucket?</a></li><li>- <a href="#">What Are the Differences Between Single-AZ and Multi-AZ Storage in OBS?</a></li><li>- <a href="#">Why Is the Message "NoSuchBucket" Displayed When I Use a User-Defined Domain Name to Access a Bucket That Can Be Accessed by the OBS Domain Name?</a></li><li>- <a href="#">What Is the Relationship Between OBS Bucket Names and Domain Names?</a></li><li>- <a href="#">Why Can't I Find the Statistics on OBS 5XX Status Codes on Cloud Eye?</a></li></ul></li></ul>

Release Date	What's New
	<ul style="list-style-type: none"> <li>- <a href="#">Will OBS Server-Side Encryption Encrypt My Existing Objects That Are Unencrypted?</a></li> </ul>
2022-10-30	<p>This issue is the seventh official release.</p> <p>This issue incorporates the following change:</p> <ul style="list-style-type: none"> <li>• Optimized the service overview and FAQs.</li> </ul>
2022-02-15	<p>This issue is the sixth official release.</p> <ul style="list-style-type: none"> <li>• Added the introduction to CTS functions in <a href="#">Cloud Trace Service</a>.</li> </ul>
2021-11-30	<p>This issue is the fifth official release.</p> <ul style="list-style-type: none"> <li>• Added FAQ <a href="#">What Redundancy Storage Techniques Does OBS Use?</a></li> <li>• Added the prerequisites for testing permissions on OBS Console in <a href="#">Overview</a> and <a href="#">Granting an IAM User Permissions to Operate a Specific Bucket</a>.</li> </ul>
2021-07-30	<p>This issue is the fourth official release.</p> <ul style="list-style-type: none"> <li>• Added descriptions about user-defined domain names.</li> </ul>
2021-04-20	<p>This issue is the third official release.</p> <ul style="list-style-type: none"> <li>• Added descriptions about server-side encryption and default bucket encryption.</li> <li>• Optimized descriptions about bucket ACLs and object ACLs.</li> <li>• Added FAQ "What Are the Differences Between OBS, EVS, and SFS?"</li> </ul>
2021-02-10	<p>This issue is the second official release.</p> <ul style="list-style-type: none"> <li>• Added the description of the parallel file system.</li> </ul>
2020-02-26	<p>This issue is the first official release.</p>