

# Alibaba Cloud Object Storage Service

## API Reference

Issue: 20190313

## Legal disclaimer

---

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

1. You shall download and obtain this document from the Alibaba Cloud website or other Alibaba Cloud-authorized channels, and use this document for your own legal business activities only. The content of this document is considered confidential information of Alibaba Cloud. You shall strictly abide by the confidentiality obligations. No part of this document shall be disclosed or provided to any third party for use without the prior written consent of Alibaba Cloud.
2. No part of this document shall be excerpted, translated, reproduced, transmitted, or disseminated by any organization, company, or individual in any form or by any means without the prior written consent of Alibaba Cloud.
3. The content of this document may be changed due to product version upgrades, adjustments, or other reasons. Alibaba Cloud reserves the right to modify the content of this document without notice and the updated versions of this document will be occasionally released through Alibaba Cloud-authorized channels. You shall pay attention to the version changes of this document as they occur and download and obtain the most up-to-date version of this document from Alibaba Cloud-authorized channels.
4. This document serves only as a reference guide for your use of Alibaba Cloud products and services. Alibaba Cloud provides the document in the context that Alibaba Cloud products and services are provided on an "as is", "with all faults" and "as available" basis. Alibaba Cloud makes every effort to provide relevant operational guidance based on existing technologies. However, Alibaba Cloud hereby makes a clear statement that it in no way guarantees the accuracy, integrity, applicability, and reliability of the content of this document, either explicitly or implicitly. Alibaba Cloud shall not bear any liability for any errors or financial losses incurred by any organizations, companies, or individuals arising from their download, use, or trust in this document. Alibaba Cloud shall not, under any circumstances, bear responsibility for any indirect, consequential, exemplary, incidental, special, or punitive damages, including lost profits arising from the use








or trust in this document, even if Alibaba Cloud has been notified of the possibility of such a loss.

5. By law, all the content of the Alibaba Cloud website, including but not limited to works, products, images, archives, information, materials, website architecture, website graphic layout, and webpage design, are intellectual property of Alibaba Cloud and/or its affiliates. This intellectual property includes, but is not limited to, trademark rights, patent rights, copyrights, and trade secrets. No part of the Alibaba Cloud website, product programs, or content shall be used, modified, reproduced, publicly transmitted, changed, disseminated, distributed, or published without the prior written consent of Alibaba Cloud and/or its affiliates. The names owned by Alibaba Cloud shall not be used, published, or reproduced for marketing, advertising, promotion, or other purposes without the prior written consent of Alibaba Cloud. The names owned by Alibaba Cloud include, but are not limited to, "Alibaba Cloud", "Aliyun", "HiChina", and other brands of Alibaba Cloud and/or its affiliates, which appear separately or in combination, as well as the auxiliary signs and patterns of the preceding brands, or anything similar to the company names, trade names, trademarks, product or service names, domain names, patterns, logos, marks, signs, or special descriptions that third parties identify as Alibaba Cloud and/or its affiliates).
6. Please contact Alibaba Cloud directly if you discover any errors in this document.



# Generic conventions

Table -1: Style conventions

| Style  | Description  | Example  |
|--|--|--|
|   | This warning information indicates a situation that will cause major system changes, faults, physical injuries, and other adverse results. |  <b>Danger:</b><br>Resetting will result in the loss of user configuration data.                                    |
|   | This warning information indicates a situation that may cause major system changes, faults, physical injuries, and other adverse results.  |  <b>Warning:</b><br>Restarting will cause business interruption. About 10 minutes are required to restore business. |
|  | This indicates warning information, supplementary instructions, and other content that the user must understand.                           |  <b>Notice:</b><br>Take the necessary precautions to save exported data containing sensitive information.          |
|  | This indicates supplemental instructions, best practices, tips, and other content that is good to know for the user.                       |  <b>Note:</b><br>You can use Ctrl + A to select all files.  |
| >  | Multi-level menu cascade.  | Settings > Network > Set network type  |
| <b>Bold</b>  | It is used for buttons, menus, page names, and other UI elements.  | Click OK.  |
| Courier font   | It is used for commands.   | Run the <code>cd / d C :/ windows</code> command to enter the Windows system folder.   |
| <i>Italics</i>   | It is used for parameters and variables.   | <code>bae log list --instanceid Instance_ID</code>   |
| [ ] or [a b]   | It indicates that it is an optional value, and only one item can be selected.  | <code>ipconfig [-all -t]</code>  |

| Style                                 | Description  | Example                             |
|---------------------------------------|--|-------------------------------------|
| <code>{}</code> or <code>{a b}</code> | It indicates that it is a required value, and only one item can be selected. | <code>switch {stand   slave}</code> |



# Contents

---

|  |     |
|--|-----|
| Legal disclaimer.....                  | I   |
| Generic conventions.....               | I   |
| 1 Access control.....                  | 1   |
| 1.1 Add a signature to the header..... | 1   |
| 1.2 Add a signature to a URL.....      | 7   |
| 1.3 Bucket access control.....         | 10  |
| 2 Service operations.....              | 12  |
| 3 Bucket operations.....               | 13  |
| 3.1 PutBucket.....                     | 13  |
| 3.2 PutBucketLogging.....              | 15  |
| 3.3 PutBucketWebsite.....              | 20  |
| 3.4 GetBucket(List Object).....        | 34  |
| 3.5 GetBucketInfo.....                 | 41  |
| 3.6 DeleteBucketWebsite.....           | 44  |
| 3.7 PutBucketLifecycle.....            | 46  |
| 4 Object operations.....               | 50  |
| 4.1 PutObject.....                     | 50  |
| 4.2 CopyObject.....                    | 56  |
| 4.3 AppendObject.....                  | 64  |
| 4.4 HeadObject.....                    | 70  |
| 4.5 PostObject.....                    | 73  |
| 4.6 Callback.....                      | 86  |
| 4.7 PutSymlink.....                    | 100 |
| 4.8 SelectObject.....                  | 102 |
| 5 Multipart upload operations.....     | 136 |
| 5.1 InitiateMultipartUpload.....       | 136 |
| 5.2 UploadPart.....                    | 140 |
| 5.3 UploadPartCopy.....                | 142 |



# 1 Access control

## 1.1 Add a signature to the header

You can add an authorization header to carry signature information in an HTTP request to indicate that the message has been authorized.

### SDK signature implementation

OSS SDK has implemented the signature. You do not need to worry about the signature issue when you use the OSS SDK. To learn more about the signature implementations of specific languages, see the OSS SDK code. The files for implementing OSS SDK signature are shown in the following table:

| SDK            | Signature implementation              |
|----------------|---------------------------------------|
| Java SDK       | <a href="#">OSSRequestSigner.java</a> |
| Python SDK     | <a href="#">auth.py</a>               |
| Net SDK        | <a href="#">OssRequestSigner.cs</a>   |
| PHP SDK        | <a href="#">OssClient.php</a>         |
| C SDK          | <a href="#">oss_auth.c</a>            |
| JavaScript SDK | <a href="#">client.js</a>             |
| Go SDK         | <a href="#">auth.go</a>               |
| Ruby SDK       | <a href="#">util.rb</a>               |
| iOS SDK        | <a href="#">OSSModel.m</a>            |
| Android SDK    | <a href="#">OSSUtils.java</a>         |

### Calculation of the Authorization field

```

Authorization = " OSS " + AccessKeyId + ":" + Signature
Signature = base64 ( hmac - sha1 ( AccessKeySecret ,
    VERB + "\ n "
    + Content - MD5 + "\ n "
    + Content - Type + "\ n "
    + Date + "\ n "
    + CanonicalizedOSSHeaders
    + CanonicalizedResource ))
  
```

- The `AccessKeySecret` indicates the key required for a signature.

- **VERB** indicates the HTTP request method, including PUT, GET, POST, HEAD, and DELETE.
- **\ n** is a line break.
- **Content - MD5** The Content-MD5 is the MD5 value of requested content data. The message content (excluding the header) is calculated to obtain an MD5 value, which is a 128-bit number. This number is encoded with Base64 into a Content-MD5 value. The request header can be used to check the message validity, that is, whether the message content is consistent with the sent content, such as “eB5eJF1ptWaXm4bijSPyxw==”. The request header may be empty. For more information, see [RFC2616 Content-MD5](#).
- **Content - Type** indicates the requested content type, such as “application/octet-stream”. Its content type may be empty.
- **Date** indicates the time that the operation takes. It must be in GMT format, such as “Sun, 22 Nov 2015 08:16:38 GMT”.
- The **CanonicalizedOSSHeaders** indicates an assembly of HTTP headers whose prefixes are “x-oss-”.
- The **CanonicalizedResource** indicates the OSS resource that the user wants to access.

Specifically, the values of Date and CanonicalizedResource cannot be empty. If the difference between the value of Date in the request and the time of the OSS server is greater than 15 minutes, the OSS server rejects the request and returns an HTTP 403 error.

### Construct CanonicalizedOSSHeaders

All the HTTP headers whose prefixes are x-oss- are called CanonicalizedOSSHeaders. The method to construct CanonicalizedResource is as follows:

1. Convert the names of all HTTP request headers whose prefixes are x-oss- into lowercase letters. For example, convert **X - OSS - Meta - Name : TaoBao** to **x - oss - meta - name : TaoBao**.
2. If the request is sent with the AccessKeyID and AccessKeySecret obtained by the STS, you must also add the obtained security-token value to the signature string in the form of **x - oss - security - token : security - token**.
3. Sort all acquired HTTP request headers in a lexicographically ascending order.

4. Delete any space on either side of a separator between the request header and content. For example, convert `x - oss - meta - name : TaoBao` to `x - oss - meta - name : TaoBao`.
5. Separate all the content and headers with the `\ n` separator to form the final CanonicalizedOSSHeaders.

**Note:**

- CanonicalizedOSSHeaders can be empty, and the `\ n` at the end can be removed.
- If only one header must be constructed, it must be `x - oss - meta - a \ n`. Note the `\ n` at the end.
- If multiple headers must be constructed, it must be `x - oss - meta - a : a \ n x - oss - meta - b : b \ n x - oss - meta - c : c \ n`. Note the `\ n` at the end.

**Construct CanonicalizedResource**

The target OSS resource specified in the request sent by the user is called a CanonicalizedResource. The method for constructing CanonicalizedResource is as follows:

1. Set CanonicalizedResource into a null character string ( "" );
2. Add the OSS resource to be accessed in the following format: `/ BucketName / ObjectName`. (If ObjectName does not exist, CanonicalizedResource is `/ BucketName/`. If BucketName does not exist either, CanonicalizedResource is `/`.)
3. If the requested resource includes sub-resources (SubResource), sort all the sub-resources in a lexicographically ascending order and separate the sub-resources using the separator `&` to generate a sub-resource string. Add `“?”` and the sub-resource string to the end of the CanonicalizedResource string. In this case, CanonicalizedResource is like: `/ BucketName / ObjectName ? acl & uploadId = UploadId`

**Note:**

- The sub-resources supported by OSS currently include: acl, uploads, location, cors, logging, website, referer, lifecycle, delete, append, tagging, objectMeta, uploadId, partNumber, security-token, position, img, style, styleName, replication, replicationProgress, replicationLocation, cname, bucketInfo, comp, qos, live,

status, vod, startTime, endTime, symlink, x-oss-process, response-content-type, response-content-language, response-expires, response-cache-control, response-content-disposition, and response-content-encoding.

- Three types of sub-resources are available:
  - Resource identifiers, such as acl, append, uploadId, and symlink sub-resources. For more information, see [Bucket-related operations](#) and [Object-related operations](#).
  - Specify response header fields such as `response -***`. For more information, see the `Request Parameters` section of [GetObject](#).
  - Object handling methods, such as `x - oss - process`. It is used as the object handling method, such as [Image Processing](#).

#### Rules to calculate a signature header

- A signature string must be in the UTF-8 format. Encode a signature string containing Chinese characters with UTF-8 first, and then use it with the AccessKeySecret to calculate the final signature.
- The signing method adopted is the HMAC-SHA1 method defined in [RFC 2104](#), where Key is `AccessKeySecret`.
- Content-Type and Content-MD5 are not required in a request. If the request requires signature verification, the null value can be replaced with the line break “\n”.
- Among all non-HTTP-standard headers, only the headers starting with “x-oss-” require signature strings, and other non-HTTP-standard headers are ignored by OSS. (For example, the “x-oss-magic” header in the preceding example must be added with a signature string.)
- Headers starting with “x-oss-” must comply with the following specifications before being used for signature verification:
  - The header name is changed to lower-case letters.
  - The headers are sorted in a lexicographically ascending order.
  - No space exists before and after the colon, which separates the header name and value.
  - Each header is followed by the line break “\n”. If no header is used, CanonicalizedOSSHeaders is set to null.

## Example signature

Assume that AccessKeyID is 44CF9590006BF252F707 and AccessKeySecret is OtxrzxIsfpFjA7SwPzILwy8Bw21TLhquhboDYROV.

| Request  | Signature string calculation formula   | Signature string   |
|--|--|--|
| PUT /nelson HTTP/1.0<br>Content-MD5: eB5eJF1ptW aXm4bijSPyxw==<br>Content-Type: text/html<br>Date: Thu, 17 Nov 2005 18:49:58 GMT<br>Host: oss-example.oss-cn-hangzhou.aliyuncs.com<br>X-OSS-Meta-Author: foo@bar.com<br>X-OSS-Magic: abracadabra | Signature = base64(hmac-sha1(AccessKeySecret, VERB + "\n" + Content-MD5 + "\n" + Content-Type + "\n" + Date + "\n" + CanonicalizedOSSHeaders + CanonicalizedResource)) | "PUT\n eB5eJF1ptW aXm4bijSPyxw==\n text/html\n Thu, 17 Nov 2005 18:49:58 GMT\n x-oss-magic: abracadabra\n x-oss-meta-author:foo@bar.com\n/oss-example/nels |

The signature calculation method is as follows:

Python sample code:

```
import base64
import hmac
import sha
h = hmac.new("OtxrzxIsfp FjA7SwPzIL wy8Bw21TLh quhboDYROV ",
             "PUT \n0DBG0ERFM DMzQTczRUY 3NUE3NzA5Q zdfNUYzMDQ\n"
             "xNEM =\n text / html \n Thu , 17 Nov 2005 18 : 49 : 58 GMT\n"
             "\n x - oss - magic : abracadabr a \n x - oss - meta - author : foo @\n"
             "bar . com \n / oss - example / nelson ", sha)
Signature = base64.b64encode(h.digest())
print("Signature : %s" % Signature)
```

The signature calculation result is 26NBxoKdsyly4EDv6inkoDft/yA=. According to the formula Authorization = "OSS " + AccessKeyID + ":" + Signature, the value of Authorization is OSS 44CF9590006BF252F707:26NBxoKdsyly4EDv6inkoDft/yA=. The value is added with the authorization header to form the message to be sent:

```
PUT /nelson HTTP/1.0
Authorization: OSS 44CF959000 6BF252F707 : 26NBxoKdsyly4EDv6inkoDft/yA=
Content-Md5: eB5eJF1ptW aXm4bijSPyxw==
Content-Type: text/html
Date: Thu, 17 Nov 2005 18:49:58 GMT
Host: oss-example.oss-cn-hangzhou.aliyuncs.com
X-OSS-Meta-Author: foo@bar.com
X-OSS-Magic: abracadabr a
```

Detail analysis are as follows:

- If the input AccessKeyID does not exist or is inactive, the error 403 Forbidden is returned. Error code: InvalidAccessKeyId.
- If the authorization value format in the user request header is incorrect, the error 400 Bad Request is returned. Error code: InvalidArgument.
- All the requests of OSS must use the GMT time format stipulated by the HTTP 1.1 protocol. Specifically, the date format is: `date1 = 2DIGIT SP month SP 4DIGIT ; day month year ( for example , 02 Jun 1982 )`. In the aforesaid date format, “day” occupies “2 digits”. Therefore, “Jun 2”, “2 Jun 1982”, and “2-Jun-82” are all invalid date formats.
- If Date is not input into the header or the format is incorrect during signature verification, the error 403 Forbidden is returned. Error code: AccessDenied.
- The request must be entered within 15 minutes based on the current time of the OSS server; otherwise, the error 403 Forbidden is returned. Error code: RequestTimeTooSkewed.
- If the AccessKeyID is active but OSS determines that the signature of the user request is incorrect, the error 403 Forbidden is returned, and the correct signature string for verification and encryption is returned to the user in the response message. The user can check whether or not the signature string is correct based on the response of OSS. Return example:

```
<? xml version = " 1 . 0 " ? >
< Error >
  < Code >
    SignatureDoesNotMatch
  </ Code >
  < Message >
    The request signature we calculated does not match the signature you provided. Check your key and signing method.
  </ Message >
  < StringToSign >
    47 45 54 0a 0a 0a 57 65 64 2c 20 31 31
    20 4d 61 79 20 32 30 31 31 20 30 37 3a
    35 39 3a 32 35 20 47 4d 54 0a 2f 75 73 72
    65 61 6c 74 65 73 74 3f 61 63 6c
  </ StringToSign >
  < RequestId >
    1E446260FF 9B10C2
  </ RequestId >
  < HostId >
    oss-cn-hangzhou.aliyuncs.com
  </ HostId >
  < SignatureProvided >
    y5H7yzPsA / tP4 + 0tH1HHvPEw Uv8 =
  </ SignatureProvided >
  < StringToSign >
    GET
    Wed , 11 May 2011 07 : 59 : 25 GMT
```

```

/ oss - example ? acl
</ StringToSi gn >
< OSSAccessK eyId >
    AKIAIVAKMS MOY7VOMRWQ
</ OSSAccessK eyId >
</ Error >

```

## Content-MD5 calculation method

Content - MD5 calculatio n  
 The message content " 123456789 " is used as an example . The Content - MD5 value of the string is calculated as follows :  
 The algorithm defined in related standards can be simplified to the following :  
 Calculate the MD5 - encrypted 128 - bit binary array .  
 Encode the binary array ( instead of the 32 - bit string code ) with Base64 .  
 Python is used as an example .  
 The correct calculatio n code is :

```

>>> import base64 , hashlib
>>> hash = hashlib . md5 ()
>>> hash . update ( " 0123456789 " )
>>> base64 . b64encode ( hash . digest () )
' eB5eJF1ptW aXm4bijSPy xw =='

```

Note :  
 The correct code is : hash . digest () , used to calculate a 128 - bit binary array

```

>>> hash . digest ()
' x \ x1e ^$] i \ xb5f \ x97 \ x9b \ x86 \ xe2 \ x8d #\ xf2 \ xc7 '

```

The common error is to base 64 the computed 32 - Bit String encoding directly .  
 An incorrect example : hash . hexdigest () , and a visible 32 - bit string is calculated .

```

>>> hash . hexdigest ()
' 781e5e245d 69b566979b 86e28d23f2 c7 '

```

Result of encoding the incorrect MD5 value with Base64 :

```

>>> base64 . b64encode ( hash . hexdigest () )
' NzgxZTVlMj Q1ZDY5YjU2 Njk3OWI4Nm UyOGQyM2Yy Yzc ='

```

## 1.2 Add a signature to a URL

In addition to using an authorization header, you can add signature information to a URL. It enables you to forward a URL to the third party for an authorized access.

### Sample code

Python sample code used to add a signature to a URL:

```

import base64
import hmac
import sha
import urllib
h = hmac . new ( " 0txrxzIsfp FjA7SwPzIL wy8Bw21TLh quhboDYROV ",
    " GET \ n \ n \ n114188912 0 \ n / oss - example / oss -
api . pdf " ,
    sha )

```

```
urllib . quote ( base64 . encodetri ng ( h . digest ( ) ) . strip ( ) )
```

OSS SDK provides the method for adding a signature into an URL. For the detailed usage, see Authorized access in the OSS SDK Reference.

To add a signature to the OSS SDK URL, see the following table.

| SDK            | URL signature method           | Implementation file            |
|----------------|--------------------------------|--------------------------------|
| Java SDK       | OSSClient.generatePresignedUrl | <a href="#">OSSClient.java</a> |
| Python SDK     | Bucket.sign_url                | <a href="#">api.py</a>         |
| Net SDK        | OssClient.GeneratePresignedUri | <a href="#">OssClient.cs</a>   |
| PHP SDK        | OssClient.signUrl              | <a href="#">OssClient.php</a>  |
| JavaScript SDK | signatureUrl                   | <a href="#">object.js</a>      |
| C SDK          | oss_gen_signed_url             | <a href="#">oss_object.c</a>   |

## Implementation

### URL signature example:

```
http :// oss - example . oss - cn - hangzhou . aliyuncs . com / oss - api . pdf ? OSSAccessK eyId = nz2pc56s93 6 ** 9l & Expires = 1141889120 & Signature = vjbyPxybdZ aNmGa % 2ByT272YEA iv4 % 3D
```

The URL signature must include at least the following three parameters: Signature, Expires, and OSSAccessK eyId.

- The Expires parameter indicates the time-out period of a URL. The value of this parameter is UNIX time (which is the number of seconds that have elapsed since 00:00:00 UTC, January 1, 1970. For more information, see [Wikipedia](#)). If the time when OSS receives the URL request is later than the value of the Expires parameter and is included in the signature, an error code request timed-out is returned. For example, if the current time is 1141889060, to create a URL that is scheduled to expire in 60 seconds, you can set the value of Expires to 1141889120. The valid period of a URL is 3,600 seconds by default and 64,800 seconds in maximum.
- OSSAccessK eyId refers to the AccessKeyID in the key.



- **Signature** indicates the signature information. For all requests and header parameters that OSS supports, the algorithm for adding a signature to a URL is basically the same as that of [Adding a signature to a header](#).

```
Signature = urlencode ( base64 ( hmac - sha1 ( AccessKeyS  ecret
'
    VERB  + "\ n "
+  CONTENT - MD5  + "\ n "
+  CONTENT - TYPE  + "\ n "
+  EXPIRES  + "\ n "
+  Canonicali zedOSSHead  ers
+  Canonicali zedResourc e )))
```

The difference is listed as follows:

- When a signature is added to a URL, the Expires parameter replaces the Date parameter.
- Signatures cannot be included in a URL and the Header at the same time.
- If more than one incoming Signature, Expires, or AccessKeyId value is available, the first of each incoming value is used.
- Whether the request time is later than the Expires time, is verified first before verifying the signature.
- When you put the signature string into a URL, remember to perform the `UrlEncode` for a URL.
- When you add a signature to a temporary user URL, the **security - token** must also be entered. The format is as follows:

```
http :// oss - example . oss - cn - hangzhou . aliyuncs . com / oss
- api . pdf ? OSSAccessK eyId = nz2pc56s93 6 ** 9l & Expires =
1141889120 & Signature = vjbyPxybdZ aNmGa % 2ByT272YEA iv4 % 3D &
security - token = SecurityTo ken
```


### Detail analysis


- If you adopt the approach of adding a signature to a URL, the authorized data is exposed on the Internet before the authorization period expires. We recommend that you must assess the usage risks in advance.
- The PUT and GET requests both support adding a signature in a URL.
- When a signature is added to a URL, the sequence of Signature, Expires, and AccessKeyId can be swapped. If one or more Signature, Expires, or AccessKeyId parameter is missing, the error 403 Forbidden is returned. Error code: AccessDenied.

- If the current access time is later than the Expires time set in the request, the error 403 Forbidden is returned. Error code: AccessDenied.
- If the format of the Expires time is incorrect, the error 403 Forbidden is returned. Error code: AccessDenied.
- If the URL includes one or more Signature, Expires, or AccessKeyId parameter and the header also includes signature information, the error 400 Bad Request is returned. Error code: InvalidArgument.
- When the signature string is generated, the Date parameter is replaced by the Expires parameter, but the headers such as content-type and content-md5 defined in the preceding section are still included. (Though the Date request header still exists in the request, you can skip adding it to the signature string.)

## 1.3 Bucket access control

OSS provides an Access Control List (ACL) for bucket-level access control. Currently, three ACLs are available for a bucket: public-read-write, public-read, and private.

| ACL               | Permission            | Description  |
|-------------------|-----------------------|--|
| public-read-write | Public read and write | <p>Any user (including anonymous users) can perform read/write operations, and delete operations on objects in the bucket.</p> <div> <b>Warning:</b><br/>We recommend you that do not set the ACL of a bucket to public-read-write to avoid incurring excessive fees or having your account suspended due to malicious or illegal activities of another user.</div> |

| ACL         | Permission                    | Description  |
|-------------|-------------------------------|--|
| public-read | Public read and private write | <p>Only the owner of the bucket can perform write operations on objects in the bucket. All other users (including anonymous users) can only perform read operations on objects in the bucket.</p> <div> <b>Warning:</b><br/>We recommend that you exercise caution when setting this ACL because it authorizes any user to perform read operations on objects in the bucket through the Internet, which may incur excessive fees.</div> |
| private     | Private read and write        | <p>Only the owner of the bucket can perform read/write operations on the objects in the bucket. Other users cannot access the objects.</p>   |

**Note:**

- If you do not set an ACL for a bucket when you create it, its ACL is set to private automatically.
- If the ACL rule of the bucket is set to private, only authorized users can access and operate on objects in the bucket. For more information about access control, see [Access control](#).
- Only the creator of an existing bucket can modify the ACL for the bucket by using the PutBucketACL API.

## 2 Service operations

---

## 3 Bucket operations

### 3.1 PutBucket

PutBucket is used to create a bucket (anonymous access is not supported).

The region where a bucket is created is consistent with the region of the endpoint from which the request is sent. Once the region of a bucket is determined, all objects in the bucket are stored in the region. For more information, see [Regions and endpoints](#).

#### Request syntax

```
PUT / HTTP/1.1
Host: BucketName.oss-cn-hangzhou.aliyuncs.com
Date: GMT Date
x-oss-acl: Permission
Authorization: SignatureValue
<?xml version="1.0" encoding="UTF-8"?>
<CreateBucketConfiguration>
  <StorageClass>Standard</StorageClass>
</CreateBucketConfiguration>
```

#### Request header

Table 3-1: Request header

|           | Type   | Description  |
|-----------|--------|--|
| x-oss-acl | String | Specifies the ACL for the bucket.<br>Valid values: public-read-write, public-read, and private |

#### Request element

Table 3-2: Request element

|              | Type   | Description  |
|--------------|--------|--|
| StorageClass | String | Specifies the storage class of the bucket.<br>Valid values: Standard, IA, and Archive. |

## Detail analysis

- The ACL for a bucket is private by default if it is not specified when the bucket is created.
- If the bucket name does not conform to the naming convention, a 400 Bad Request message is returned with an error code: InvalidBucketName.
- If user authentication information is not carried in a PutBucket request, a 403 Forbidden message is returned with an error code: AccessDenied.
- You can create up to 30 buckets within the same region. If more than 30 buckets are created within a region, a 400 Bad Request message is returned with an error code: TooManyBuckets.
- When creating a bucket, you can specify the data redundancy type (DataRedundancyType) of the bucket. The valid values include: LRS (local data redundancy, default value) and ZRS (zone redundancy).

## Examples

### Request example:

```
PUT / HTTP/1.1
Host: oss-example.oss-cn-hangzhou.aliyuncs.com
Date: Fri, 24 Feb 2017 03:15:40 GMT
x-oss-acl: private
Authorization: OSS qn6qrrqxo2 oawuk53otf jbyc:77Dvh5wQgIjWjw0/KyRt8dOPfo8=
<?xml version="1.0" encoding="UTF-8"?>
<CreateBucketConfiguration>
  <StorageClass>Standard</StorageClass>
</CreateBucketConfiguration>
```

### Response example:

```
HTTP/1.1 200 OK
x-oss-request-id: 534B371674E88A4D8906008B
Date: Fri, 24 Feb 2017 03:15:40 GMT
Location: /oss-example
Content-Length: 0
Connection: keep-alive
```

Server : AliyunOSS

## 3.2 PutBucketLogging

Bucket owners can use PutBucketLogging to enable the access logging function for their bucket.

When this function is enabled, OSS automatically records the details about the requests to this bucket, and follows the user-specified rules to write the access logs as an object into a user-specified bucket on an hourly basis.



### Note:

OSS provides bucket access logs for bucket owners to understand and analyze bucket access behaviors easily. The bucket access logs provided by OSS do not guarantee that every single access record is logged.

### Request syntax

```
PUT /? logging HTTP / 1 . 1
Date : GMT Date
Content - Length : ContentLength
Content - Type : application / xml
Authorization : SignatureValue
Host : BucketName . oss - cn - hangzhou . aliyuncs . com
<? xml version =" 1 . 0 " encoding =" UTF - 8 "? >
< BucketLoggingStatus >
  < LoggingEnabled >
    < TargetBucket > TargetBucket </ TargetBucket >
    < TargetPrefix > TargetPrefix </ TargetPrefix >
  </ LoggingEnabled >
</ BucketLoggingStatus >
```

### Request elements

| Name                | Type      | Required | Description  |
|---------------------|-----------|----------|--|
| BucketLoggingStatus | container | Yes      | The container for storing access log status information<br>Child element: LoggingEnabled<br>Parent element: none   |
| LoggingEnabled      | container | No       | The container for storing access log information. This element is required only when server access logging is enabled.<br>Child element: TargetBucket, TargetPrefix<br>Parent element: BucketLoggingStatus |

| Name         | Type      | Required   | Description   |
|--------------|-----------|--|---|
| TargetBucket | character | This element is required when server access logging is enabled | The bucket for storing access logs.<br>Child element: none<br>Parent element: BucketLoggingStatus.<br>LoggingEnabled                |
| TargetPrefix | character | No   | The prefix of the names of saved access log files.<br>Child element: none<br>Parent element: BucketLoggingStatus.<br>LoggingEnabled |

### Naming rules for the objects storing access logs

```
< TargetPrefix >< SourceBucket >- YYYY - mm - DD - HH - MM - SS - UniqueString
```

In the naming rules, the TargetPrefix is specified by the user; YYYY, mm, DD, HH, MM, and SS give the year, month, day, hour, minutes, and seconds of the creation time in Arabic numerals (note the digits); and UniqueString is the string generated by OSS system. An example for the name of an object actually used to store OSS access logs is given as follows:

```
MyLog - oss - example - 2012 - 09 - 10 - 04 - 00 - 00 - 0000
```

In the preceding example, “MyLog- “ is the Object prefix specified by the user; “oss-example” is the name of the origin bucket; “2012-09-10-04-00-00” is the Object creation time (Beijing time); and “0000” is the string generated by OSS system.

### Log file format

| Name      | Example                      | Description  |
|-----------|------------------------------|--|
| Remote IP | 119.140.142.11               | IP address from which the request is initiated (the proxy or user firewall may block this field) |
| Reserved  | -                            | Reserved field   |
| Reserved  | -                            | Reserved field   |
| Time      | [02/May/2012:00:00:04 +0800] | Time when OSS receives the request   |



| Name                  | Example                                     | Description   |
|-----------------------|---|---|
| Request-URI           | “GET /aliyun-logo.png HTTP/1.1 “            | User-Requested URI (including query-string)                       |
| HTTP Status           | 200   | HTTP status code returned by OSS                                  |
| SentBytes             | 5576  | Traffic that the user downloads from OSS                          |
| RequestTime (ms)      | 71  | Time utilized in completing this request (in ms)                  |
| Referer               | http :// www . aliyun . com / product / oss | HTTP Referer in the request                                       |
| User-Agent            | curl/7.15.5                                 | HTTP User-Agent header  |
| HostName              | oss-example.regionid.example.com            | Domain name for access request                                    |
| Request ID            | 505B01695037C2AF032593A4                    | UUID used to uniquely identify this request                       |
| LoggingFlag           | true  | Whether the access logging function is enabled                    |
| Requester Aliyun ID   | 1657136103983691                            | Alibaba Cloud ID of the requester, “- “ for an anonymous access   |
| Operation             | GetObject                                   | Request type  |
| Bucket                | oss-example                                 | Name of the bucket requested for access                           |
| Key                   | /aliyun-logo.png                            | Key of user request   |
| ObjectSize            | 5576  | Object size   |
| Server Cost Time (ms) | 17  | Time utilized by OSS server to process this request (in ms)       |
| Error Code            | NoSuchBucket                                | Error code returned by OSS  |
| Request Length        | 302   | Length of user request (byte)                                     |
| UserID                | 1657136103983691                            | ID of the bucket owner  |
| Delta DataSize        | 280   | Bucket size variation, “- “ for no change                         |
| Sync Request          | -   | Whether this is an origin retrieval request from CDN, “- “ for no |
| Reserved              | -   | Reserved field  |

## Detail analysis

- The source bucket and target bucket must belong to the same user.
- In the preceding request syntax, “BucketName” refers to the bucket for which access logging is enabled; “TargetBucket” refers to the bucket into which access logs are saved; “TargetPrefix” refers to the name prefix of the object storing access logs and can be null.
- The source bucket and target bucket can be the same or different buckets. You can save logs from multiple source buckets to the same target bucket (in this case, we recommend that you assign different values to TargetPrefix).
- To disable the access logging function for a bucket, you only must send an empty BucketLoggingStatus request. For a detailed method, see the following request example.
- All PUT Bucket Logging requests must be provided with signatures, because the anonymous access is not supported.
- If the initiator of a PUT Bucket Logging request is not the owner of the source bucket (BucketName in the request example), OSS returns error code 403.
- If the source bucket does not exist, OSS returns the error code: NoSuchBucket.
- If the initiator of a PUT Bucket Logging request is not the owner of the target bucket (indicated by TargetBucket in the request example), OSS returns Error 403. If the target bucket does not exist, OSS returns the error code: InvalidTargetBucketForLogging.
- The source bucket and target bucket must belong to the same data center. Otherwise, Error 400 with the error code: InvalidTargetBucketForLogging is returned.
- If a PUT Bucket Logging request has an invalid XML, the error code: MalformedXML is returned.
- The source bucket and target bucket can be the same bucket. You can save the logs of different source buckets into the same target bucket (note that you must set TargetPrefix to different values).
- When the source bucket is deleted, the corresponding logging rules are also deleted.
- OSS generates a bucket access log file every hour. However, all requests during the hour may not be recorded in the log file, but may get recorded in the previous or next log file.

- In the naming rules for log files generated by OSS, “UniqueString” is only a UUID that OSS generates for a file to uniquely identify the file.
- Each time OSS generates a bucket access log file, this is considered a PUT operation and the occupied space is recorded, but the generated traffic is not recorded. After log files are generated, you can operate these log files as common objects.
- OSS ignores all query-string parameters prefixed by “x- “ but such query-string parameters are recorded in access logs. If you want to mark a special request from massive access logs, you can add a query-string parameter prefixed by “x- “ to the URL. For example:

```
http://oss-example.oss-cn-hangzhou.aliyuncs.com/aliyun-logo.png
```

```
http://oss-example.regionid.example.com/aliyun-logo.png?x-user=admin
```

- When OSS processes the preceding two requests, the results are the same. However, you can search access logs with “x-user=admin” to quickly locate the marked request.
- You may see “- “ in any field of OSS logs. It indicates that data is unknown or the field is invalid for the current request.
- Certain fields are added to the end of OSS log files in future based on the requirements. We recommend that developers to consider compatibility issues when developing log processing tools.
- If you have uploaded the Content-MD5 request header, OSS calculates the body’s Content-MD5 and checks if the two are the same. If the two are different, the error code: InvalidDigest is returned.

## Example

Example of a request for enabling bucket access logging:

```
PUT /? logging HTTP/1.1
Host: oss-example.oss-cn-hangzhou.aliyuncs.com
Content-Length: 186
Date: Fri, 04 May 2012 03:21:12 GMT
Authorization: OSS qn6qrrqxo2 oawuk53otf jbyc: KU5h8YMUC78M30dXqf3J xrTZhIA =
<? xml version="1.0" encoding="UTF-8" ?>
< BucketLoggingStatus >
< LoggingEnabled >
< TargetBucket > doc-log </ TargetBucket >
< TargetPrefix > MyLog -</ TargetPrefix >
</ LoggingEnabled >
```

```
</ BucketLogg ingStatus >
```

#### Response example:

```
HTTP / 1 . 1 200 OK
x - oss - request - id : 534B371674 E88A4D8906 008B
Date : Fri , 04 May 2012 03 : 21 : 12 GMT
Content - Length : 0
Connection : keep - alive
Server : AliyunOSS
```

#### Example of a request for disabling bucket access logging:

```
PUT /? logging HTTP / 1 . 1
Host : oss - example . oss - cn - hangzhou . aliyuncs . com
Content - Type : applicatio n / xml
Content - Length : 86
Date : Fri , 04 May 2012 04 : 21 : 12 GMT
Authorizat ion : OSS qn6qrrqxo2 oawuk53otf jbyc : KU5h8YMUC7
8M30dXqf3J xrTzHiA =
<? xml version =" 1 . 0 " encoding =" UTF - 8 "? >
< BucketLogg ingStatus >
</ BucketLogg ingStatus >
```

#### Response example:

```
HTTP / 1 . 1 200 OK
x - oss - request - id : 534B371674 E88A4D8906 008B
Date : Fri , 04 May 2012 04 : 21 : 12 GMT
Content - Length : 0
Connection : keep - alive
Server : AliyunOSS
```

## 3.3 PutBucketWebsite

PutBucketWebsite is used to set a bucket to static website hosting mode and set routing rules.

#### website

The website interface provides the following two features:

- Sets the default home page and the default 404 page.
- Sets the RoutingRule. The RoutingRule is used to specify the 3xx routing rules and mirroring back-to-origin rules.



#### Note:

Mirroring back-to-origin is supported in Alibaba Cloud and Finance Cloud.

The following example shows the fields of website:

```
< WebsiteCon figuration >
```

```

< IndexDocum ent >
  < Suffix > index . html </ Suffix >
</ IndexDocum ent >
< ErrorDocum ent >
  < Key > error . html </ Key >
</ ErrorDocum ent >
< RoutingRul es >
  < RoutingRul e >
    < RuleNumber > 1 </ RuleNumber >
    < Condition >
      < KeyPrefixE quals > abc </ KeyPrefixE quals >
      < HttpErrorC odeReturne dEquals > 404 </ HttpErrorC
odeReturne dEquals >
    </ Condition >
    < Redirect >
      < RedirectTy pe > Mirror </ RedirectTy pe >
      < PassQueryS tring > true </ PassQueryS tring >
      < MirrorURL > http :// www . test . com </ MirrorURL >
      < MirrorPass QueryStrin g > true </ MirrorPass QueryStrin
g >
      < MirrorFoll owRedirect > true </ MirrorFoll owRedirect >
      < MirrorChec kMd5 > false </ MirrorChec kMd5 >
      < MirrorHead ers >
        < PassAll > true </ PassAll >
        < Pass > myheader - key1 </ Pass >
        < Pass > myheader - key2 </ Pass >
        < Remove > myheader - key3 </ Remove >
        < Remove > myheader - key4 </ Remove >
        < Set >
          < Key > myheader - key5 </ Key >
          < Value > myheader - value5 </ Value >
        </ Set >
      </ MirrorHead ers >
    </ Redirect >
  </ RoutingRul e >
  < RoutingRul e >
    < RuleNumber > 2 </ RuleNumber >
    < Condition >
      < KeyPrefixE quals > abc </ KeyPrefixE quals >
      < HttpErrorC odeReturne dEquals > 404 </ HttpErrorC
odeReturne dEquals >
      < IncludeHea der >
        < Key > host </ Key >
        < Equals > test . oss - cn - beijing - internal . aliyuncs .
com </ Equals >
      </ IncludeHea der >
    </ Condition >
    < Redirect >
      < RedirectTy pe > AliCDN </ RedirectTy pe >
      < Protocol > http </ Protocol >
      < HostName > www . test . com </ HostName >
      < PassQueryS tring > false </ PassQueryS tring >
      < ReplaceKey With > prefix /${ key }. suffix </ ReplaceKey
With >
      < HttpRedire ctCode > 301 </ HttpRedire ctCode >
    </ Redirect >
  </ RoutingRul e >
</ RoutingRul es >
</ WebsiteCon figuration >

```

## Request syntax

```
PUT /? website HTTP / 1 . 1
```

```

Date : GMT Date
Content - Length : ContentLength
Content - Type : application / xml
Host : BucketName . oss - cn - hangzhou . aliyuncs . com
Authorization : SignatureValue

<? xml version =" 1 . 0 " encoding =" UTF - 8 "? >
< WebsiteConfiguration >
  < IndexDocument >
    < Suffix > index . html </ Suffix >
  </ IndexDocument >
  < ErrorDocument >
    < Key > errorDocument . html </ Key >
  </ ErrorDocument >
</ WebsiteConfiguration >

```

### Request elements

| Name                 | Type      | Description   | Required   |
|----------------------|-----------|---|--|
| WebsiteConfiguration | Container | Root element<br>Parent element: None  | Yes  |
| IndexDocument        | Container | Specifies the container for the default home page.<br>Parent element: WebsiteConfiguration  | Conditionally required. You must specify at least one of the following containers: IndexDocument, ErrorDocument, and RoutingRules. |
| Suffix               | String    | Specifies the default home page.<br>If this element is configured, access to an object with a slash (/) at the end of its name is redirected to the default home page.<br>Parent element: IndexDocument | Conditionally required. This element must be specified when its parent element IndexDocument is specified.                         |

|               |                  |   |  |
|---------------|------------------|---|--|
| ErrorDocument | Container        | Specifies the container for the 404 page.<br>Parent element: WebsiteConfiguration   | Conditionally required. You must specify at least one of the following containers: IndexDocument, ErrorDocument, and RoutingRules. |
| Key           | Container        | 404 page<br>If this element is specified, access to an object that does not exist is redirected to the 404 page.<br>Parent element: ErrorDocument   | Conditionally required. This element must be specified when its parent element ErrorDocument is specified.                         |
| RoutingRules  | Container        | Specifies the container for the RoutingRule.<br>Parent element: WebsiteConfiguration  | Conditionally required. You must specify at least one of the following containers: IndexDocument, ErrorDocument, and RoutingRules. |
| RoutingRule   | Container        | Specifies routing rules or mirroring back-to-origin rules. You can specify a maximum of five RoutingRules.<br>Parent element: RoutingRules  | No   |
| RuleNumber    | Positive integer | Specifies the sequence number used to match and execute routing rules. Routing rules are matched according to the sequence numbers. If a routing rule matches the number, the rule is executed and the following rules are not executed.<br>Parent element: RoutingRule | Conditionally required. This element must be specified when its parent element RoutingRule is specified.                           |

|                             |                  |  |  |
|-----------------------------|------------------|--|--|
| Condition                   | Container        | Specifies the matching conditions. If a routing rule meets all the conditions, it is executed. The elements in the bucket are in the AND relationship, that is, a routing rule must meet all the conditions before it can be considered matched .<br>Parent element: RoutingRule | Conditionally required. This element must be specified when its parent element RoutingRule is specified.   |
| KeyPrefixEquals             | String           | Indicates that only objects that match the prefix can match the rule.<br>Parent element: Condition   | No   |
| HttpErrorCodeReturnedEquals | HTTP status code | Indicates that the rule can be matched only when the object returns the specified status code when being accessed. If the routing rule is a mirroring back-to-source rule, this status code must be 404.<br>Parent element: Condition  | No   |
| IncludeHeader               | Container        | Indicates that the routing rule can be matched only when the specified header is included in the request and the header value equals the specified value. You can specify a maximum five of the same container .<br>Parent element: Condition                                    | No   |
| Key                         | String           | Indicates that the rule is matched only when this header is included in the request and the header value equals the value specified by Equals.<br>Parent element: IncludeHeader  | Conditionally required. This element must be specified when its parent element IncludeHeader is specified. |



|              |           |   |  |
|--------------|-----------|---|--|
| Equals       | String    | Indicates that the rule can be matched only when the header specified by Key is included in the request and the header value equals to the specified value.<br>Parent element: IncludeHeader  | Conditionally required. This element must be specified when its parent element IncludeHeader is specified. |
| Redirect     | Container | Specifies the actions to perform after the rule is matched.<br>Parent element: RoutingRule  | Conditionally required. This element must be specified when its parent element RoutingRule is specified.   |
| RedirectType | String    | Specifies the redirecting type, which has the following available values: <ul style="list-style-type: none"> <li>• Mirror (mirroring back-to-origin)</li> <li>• External (external redirection, that is, OSS returns a 3xx request which redirects the access to another IP address.)</li> <li>• Internal (internal redirection, that is, OSS redirects the access from object1 to object2 based on the rule. In this case, the user accesses object2 but not object1.)</li> <li>• AliCDN (AliCDN redirection, which is used for AliCDN. Unlike the External type, OSS adds an additional header to the request . After identifying the header, AliCDN redirects the access to the specified IP address and returns the obtained data but not the 3xx redirecting request to the user.)</li> </ul> Parent element: Redirect | Conditionally required. This element must be specified when its parent element Redirect is specified.      |

|                 |        |  |   |
|-----------------|--------|--|---|
| PassQueryString | Bool   | <p>Indicates whether the request parameter is carried when the redirection or mirroring back-to-origin is performed. The available value of the element is true or false. For example, if the parameter "?a=b&amp;c=d" is carried in a request to OSS and this element is set to true, this parameter is added to the Location header when the rule is 302 redirection. For example, if the request is "Location:www.test.com?a=b&amp;c=d" and the redirecting type is mirroring back-to-origin, the parameter is also carried in the back-to-origin request.</p> <p>Default value: false</p> <p>Parent element: Redirect</p>  | No  |
| MirrorURL       | String | <p>Indicates the IP address of the origin site in the mirroring back-to-origin. This element takes effect only when the value of RedirectType is Mirror. If the MirrorURL starts with http:// or s://, it must be ended with a slash (/). OSS constructs the back-to-origin URL by adding the target object to the MirrorURL. For example, if MirrorURL is set to <code>http://www.test.com</code> and the object to be accessed is "myobject", the back-to-origin URL is <code>http://www.test.com/dir1/myobject</code>. If MirrorURL is set to <code>http://www.test.com/dir1/</code>, the back-to-origin URL is <code>http://www.test.com/dir1/myobject</code>.</p> <p>Parent element: Redirect</p> | Conditionally required. This element must be specified if the RedirectType is Mirror. |

|                           |      |   |    |
|---------------------------|------|---|----|
| MirrorPass<br>QueryString | Bool | <p>This element plays the same role as PassQueryString and has a higher priority than PassQueryString. However, this element take effects only when the RedirectType is Mirror.</p> <p>Default value: false</p> <p>Parent element: Redirect</p>   | No |
| MirrorFollowRedirect      | Bool | <p>Indicates whether the access is redirected to the specified Location if the origin site returns a 3xx status code when receiving a back-to-origin request.</p> <p>For example, the origin site returns a 302 status code and specifies the Location when receiving a mirroring back-to-origin request. In this case, if the value of MirrorFollowRedirect is true, OSS continues to send requests to the IP address specified by the Location. (A request can be redirected for a maximum of 10 times. If the request is redirected for more than 10 times, a mirroring back-to-origin failure message is returned.) If the value of MirrorFollowRedirect is false, OSS returns a 302 status code and passes through the Location. This element takes effect only when the value of RedirectType is Mirror.</p> <p>Default value: true</p> <p>Parent element: Redirect</p> | No |

|                |           |  |    |
|----------------|-----------|--|----|
| MirrorCheckMd5 | Bool      | Indicates whether OSS performs an MD5 check on the body of the response returned by the origin site. When the value of this element is true and the response returned by the origin site includes a Content-Md5 header, OSS checks whether the MD5 checksum of the obtained data matches the header. If not, OSS does not store the data. This element takes effect only when the value of RedirectType is Mirror.<br>Default value: false<br>Parent element: Redirect | No |
| MirrorHeaders  | Container | Specifies the header carried in the response returned by the origin site. This element takes effect only when the value of RedirectType is Mirror.<br>Parent element: Redirect   | No |
| PassAll        | Bool      | Indicates whether OSS passes through all headers (except for reserved headers and the headers starting with oss-/x-oss-/x-drs-) to the origin site. This element takes effect only when the value of RedirectType is Mirror.<br>Default value: false<br>Parent element: MirrorHeaders  | No |
| Pass           | String    | Specifies the headers that are passed through to the origin site. A maximum of 10 headers can be specified. The maximum length of a header is 1,024 bytes. The character set of this element is: 0-9, A-Z, a-z, and dash. This element takes effect only when the value of RedirectType is Mirror.<br>Parent element: MirrorHeaders  | No |

|        |           |   |  |
|--------|-----------|---|--|
| Remove | String    | Specifies the headers that cannot be passed to the origin site. A maximum of 10 headers can be specified (including repeated headers). This element is used together with PassAll. The maximum length of a header is 1,024 bytes. The character set of this element is the same as that of Pass. This element takes effect only when the value of RedirectType is Mirror. Parent element: MirrorHeaders | No   |
| Set    | Container | Specifies headers that are sent to the origin site. The specified headers are configured in the data returned by the origin site no matter whether they are carried in the request. A maximum of 10 groups of headers can be configured (including repeated headers). This element takes effect only when the value of RedirectType is Mirror. Parent element: MirrorHeaders                            | No   |
| Key    | String    | Specifies the key of the header. The maximum length of a key is 1,024 bytes. The character set of this element is the same as that of Pass. This element takes effect only when the value of RedirectType is Mirror. Parent element: Set  | Conditionally required. This element must be specified when its parent element Set is specified. |
| Value  | String    | Specifies the value of the header. The maximum length of the value is 1,024 bytes. The character "\r\n" is not allowed in the element. This element takes effect only when the value of RedirectType is Mirror. Parent element: Set   | Conditionally required. This element must be specified when its parent element Set is specified. |

|                  |                  |  |  |
|------------------|------------------|--|--|
| Protocol         | String           | Indicates the protocol used for redirections. The available value of this element is http or https. For example, the Location header is <code>https :// www . test . com / test</code> if the requested object is test, the request is redirected to <code>www . test . com</code> , and the value of Protocol is https. This element takes effect only when the value of RedirectType is External or AliCDN.<br>Parent element: Redirect          | Conditionally required. This element must be specified when the value of RedirectType is not External or AliCDN. |
| HostName         | String           | Indicates the domain name used for redirections, which must comply with the specifications for domain names. For example, the Location header is <code>https :// www . test . com / test</code> if the requested object is test, the value of Protocol is https, and the Hostname is specified to <code>www . test . com</code> . This element takes effect only when the value of RedirectType is External or AliCDN.<br>Parent element: Redirect | Conditionally required. This element must be specified when the value of RedirectType is not External or AliCDN. |
| HttpRedirectCode | HTTP status code | Indicates the returned status code in redirections. The available value of this element is 301, 302, or 307. This element takes effect only when the value of RedirectType is External or AliCDN.<br>Parent element: Redirect  | Conditionally required. This element must be specified when the value of RedirectType is not External or AliCDN. |

|                          |        |   |  |
|--------------------------|--------|---|--|
| ReplaceKey<br>PrefixWith | String | <p>Indicates the string used to replace the prefix of the requested object name in redirections. If the prefix of the object name is empty, this string is added before the object name. The ReplaceKeyWith and ReplaceKeyPrefixWith elements cannot be set simultaneously.</p> <p>For example, if KeyPrefixEquals is set to abc/ and ReplaceKeyPrefixWith is set to def/, the Location header for an object named abc/test.txt is <code>http://www.test.com/def/test.txt</code>. This element takes effect only when the value of RedirectType is External or AliCDN.</p> <p>Parent element: Redirect</p>          | Conditionally required. This element must be specified when the value of RedirectType is not External or AliCDN. |
| ReplaceKey<br>With       | String | <p>Indicates the string used to replace the requested object name in redirections. This element can be a variable. (The <code>\${key}</code> variable indicating the object name in the request is supported.) The ReplaceKeyWith and ReplaceKeyPrefixWith elements cannot be set simultaneously.</p> <p>For example, if ReplaceKeyWith is set to <code>prefix/\${key}.suffix</code>, the Location header for an object named test is <code>http://www.test.com/prefix/test.suffix</code>. This element takes effect only when the value of RedirectType is External or AliCDN.</p> <p>Parent element: Redirect</p> | Conditionally required. This element must be specified when the value of RedirectType is not External or AliCDN. |

#### Detail Analysis

- Static websites are the websites where all Web pages are composed of static content, including scripts such as JavaScript executed on the client. OSS does not support content that needs to be processed by the server, such as PHP, JSP, and APS.NET.

- To use your own domain name to access bucket-based static websites, you can use the CNAME. For more information about the configuration method, see [Bind custom domain names \(CNAME\)](#).
- To set a bucket to static website hosting mode, you must specify the index page, and the error page is optional.
- To set a bucket to static website hosting mode, the specified index page and error page are objects in the bucket.
- After a bucket is set to static website hosting mode, OSS returns the index page for anonymous access to the root domain name of the static website, and returns the results of Get Bucket for signed access to the root domain name of the static website.
- If you upload the Content-MD5 request header, OSS calculates the body's Content-MD5 and checks whether the two values are the same. If the two values are different, an InvalidDigest error code is returned.

## Example

### Request example:

```
PUT /? website HTTP / 1 . 1
Host : oss - example . oss - cn - hangzhou . aliyuncs . com
Content - Length : 209
Date : Fri , 04 May 2012 03 : 21 : 12 GMT
Authorization : OSS qn6qrrqxo2 oawuk53otf jbyc : KU5h8YMUC7
8M30dXqf3J xrTzHiA =

<? xml version =" 1 . 0 " encoding =" UTF - 8 "? >
< WebsiteCon figuration >
< IndexDocum ent >
< Suffix > index . html </ Suffix >
</ IndexDocum ent >
< ErrorDocum ent >
< Key > error . html </ Key >
</ ErrorDocum ent >
</ WebsiteCon figuration >
```

### Response example:

```
HTTP / 1 . 1 200 OK
x - oss - request - id : 534B371674 E88A4D8906 008B
Date : Fri , 04 May 2012 03 : 21 : 12 GMT
Content - Length : 0
Connection : keep - alive
Server : AliyunOSS
```

### Complete code:

```
PUT /? website HTTP / 1 . 1
Date : Fri , 27 Jul 2018 09 : 03 : 18 GMT
```



```

Content - Length : 2064
Host : test . oss - cn - hangzhou - internal . aliyuncs . com
Authorizat ion : OSS alnBNgkzzx cQMf8u : sNKIHT6ci / z231yIT5vY
netDLu4 =
User - Agent : aliyun - sdk - python - test / 0 . 4 . 0

< WebsiteCon figuration >
< IndexDocum ent >
< Suffix > index . html </ Suffix >
</ IndexDocum ent >
< ErrorDocum ent >
< Key > error . html </ Key >
</ ErrorDocum ent >
< RoutingRul es >
< RoutingRul e >
< RuleNumber > 1 </ RuleNumber >
< Condition >
< KeyPrefixE quals > abc </ KeyPrefixE quals >
< HttpErrorC odeReturne dEquals > 404 </ HttpErrorC odeReturne
dEquals >
</ Condition >
< Redirect >
< RedirectTy pe > Mirror </ RedirectTy pe >
< PassQueryS tring > true </ PassQueryS tring >
< MirrorURL > http :// www . test . com </ MirrorURL >
< MirrorPass QueryStrin g > true </ MirrorPass QueryStrin g >
< MirrorFoll owRedirect > true </ MirrorFoll owRedirect >
< MirrorChec kMd5 > false </ MirrorChec kMd5 >
< MirrorHead ers >
< PassAll > true </ PassAll >
< Pass > myheader - key1 </ Pass >
< Pass > myheader - key2 </ Pass >
< Remove > myheader - key3 </ Remove >
< Remove > myheader - key4 </ Remove >
< Set >
< Key > myheader - key5 </ Key >
< Value > myheader - value5 </ Value >
</ Set >
</ MirrorHead ers >
</ Redirect >
</ RoutingRul e >
< RoutingRul e >
< RuleNumber > 2 </ RuleNumber >
< Condition >
< KeyPrefixE quals > abc </ KeyPrefixE quals >
< HttpErrorC odeReturne dEquals > 404 </ HttpErrorC odeReturne
dEquals >
< IncludeHea der >
< Key > host </ Key >
< Equals > test . oss - cn - beijing - internal . aliyuncs . com </
Equals >
</ IncludeHea der >
</ Condition >
< Redirect >
< RedirectTy pe > AliCDN </ RedirectTy pe >
< Protocol > http </ Protocol >
< HostName > www . test . com </ HostName >
< PassQueryS tring > false </ PassQueryS tring >
< ReplaceKey With > prefix /${ key }. suffix </ ReplaceKey With >
< HttpRedire ctCode > 301 </ HttpRedire ctCode >
</ Redirect >
</ RoutingRul e >
</ RoutingRul es >
</ WebsiteCon figuration >

```

```

HTTP / 1 . 1    200    OK
Server : AliyunOSS
Date : Fri , 27 Jul 2018 09 : 03 : 18 GMT
Content - Length : 0
Connection : keep - alive
x - oss - request - id : 5B5ADFD6ED 3CC49176CB E29D
x - oss - server - time : 47

```

## 3.4 GetBucket(List Object)

The GetBucket operation can be used to list all of the object information in a bucket.

### Request syntax

```

GET / HTTP / 1 . 1
Host : BucketName . oss - cn - hangzhou . aliyuncs . com
Date : GMT Date
Authorizat ion : SignatureV alue

```

### Request parameters

When you initiate a GetBucket (ListObject) request, you can use prefix, marker, delimiter, and max-keys to prescribe a limit to the list to return partial results. Besides, encoding-type can be used to encode the following elements in the returned results: delimiter, marker, prefix, NextMarker, and key.

| Name       | Data type | Required | Description   |
|------------|-----------|----------|---|
| delimiter  | string    | No       | A character used to group object names. All the names of the objects that contain a specified prefix and after which the delimiter occurs for the first time, act as a group of elements - CommonPrefixes.<br>Default value: None |
| marker     | string    | No       | Sets the returned results to begin from the first entry after the marker in alphabetical order.<br>Default value: None  |
| max - keys | string    | No       | Limits the maximum number of objects returned for one request. If not specified, the default value is 100. The max-keys value cannot exceed 1000.<br>Default value: 100   |
| prefix     | string    | No       | Limits that the returned object key must be prefixed accordingly. Note that the keys returned from queries using a prefix still contain the prefix.<br>Default value: None  |

| Name               | Data type | Required | Description  |
|--------------------|-----------|----------|--|
| encoding<br>- type | string    | No       | Specifies the encoding of the returned content and the encoding type. Parameters delimiter, marker, prefix, NextMarker, and key use UTF-8 characters, but the XML 1.0 Standard does not support parsing certain control characters, such as characters with ASCII values ranging from 0 to 10. If some elements in the returned results contain characters that are not supported by the XML 1.0 Standard, encoding-type can be specified to encode these elements, such as delimiter, marker, prefix, NextMarker, and key. Default value: None; Optional value: URL |

## Response elements

| Name           | Type      | Description  |
|----------------|-----------|--|
| Contents       | container | Container used for saving every returned object meta.<br>Parent node: ListBucketResult   |
| CommonPrefixes | string    | If the delimiter parameter is specified in the request, the response returned by OSS contains the CommonPrefixes element. This element indicates the set of objects which ends with a delimiter and have a common prefix.<br>Parent node: ListBucketResult |
| Delimiter      | string    | A character used to group object names. All those objects whose names contain the specified prefix and after which the delimiter occurs for the first time, act as a group of elements - CommonPrefixes.<br>Parent node: ListBucketResult                  |
| EncodingType   | string    | Encoding type for the returned results. If encoding-type is specified in a request, the following elements in the returned results are encoded: delimiter, marker, prefix, NextMarker, and key.<br>Parent node: ListBucketResult                           |
| DisplayName    | string    | Name of the object owner.<br>Parent node: ListBucketResult.Contents.Owner  |

| Name              | Type                 | Description   |
|-------------------|----------------------|---|
| ETag              | string               | The ETag (entity tag) is created when an object is generated and is used to indicate the content of the object. For an object created by a Put Object request, the value of ETag is the value of MD5 in the content of the object. For an object created in other way, the value of ETag is the UUID in the content of the object. The value of ETag can be used to check whether the content of the object is changed. We recommend that the ETag be used as the MD5 value of the object content to verify data integrity.<br>Parent node: ListBucketResult.Contents |
| ID                | string               | User ID of the bucket owner.<br>Parent node: ListBucketResult.Contents.Owner  |
| IsTruncated       | enumerated<br>string | Indicates whether all results have been returned; “true” means that not all results are returned this time; “false” means that all results are returned this time.<br>Valid values: true and false<br>Parent node: ListBucketResult   |
| Key               | string               | Key of an object<br>Parent node: ListBucketResult.Contents  |
| LastModified      | time                 | The latest modification time of an object.<br>Parent node: ListBucketResult.Contents  |
| ListBucket Result | container            | Container for storing the results of the “Get Bucket” request<br>subnodes: Name, Prefix, Marker, MaxKeys, Delimiter, IsTruncated, Nextmarker, and Contents<br>Parent node: None   |
| Marker            | string               | Marks the origin of the current Get Bucket (List Object) request.<br>Parent node: ListBucketResult  |
| MaxKeys           | string               | The maximum number of returned results in response to the request.<br>Parent node: ListBucketResult   |
| Name              | string               | Name of a bucket<br>Parent node: ListBucketResult   |

| Name         | Type      | Description  |
|--------------|-----------|--|
| Owner        | container | Container used for saving the information about the bucket owner.<br>subnodes: DisplayName and ID<br>Parent node: ListBucketResult   |
| Prefix       | string    | Starting prefix for the current results of query.<br>Parent node: ListBucketResult   |
| Size         | string    | Number of bytes of the object.<br>Parent node: ListBucketResult.Contents   |
| StorageClass | string    | Indicates Object storage type. “Standard” , “IA” , and “Archive” types are available. (Currently, the “Archive” type is only available in some regions.)<br>Parent node: ListBucketResult.Contents |

#### Detail analysis

- The custom meta in the object is not returned during the GetBucket request.
- If the bucket to be accessed does not exist, or if you attempt to access a bucket which cannot be created because of standard naming rules are not followed when naming a bucket, Error 404 Not Found with the error code “NoSuchBucket” is returned.
- If you have no permission to access the bucket, the system returns Error 403 Forbidden with the error code “AccessDenied” .
- If listing cannot be completed at one time because of the max-keys setting, a `<NextMarker >` is appended to the returned result, prompting that this can be taken as a marker for continued listing. The value in NextMarker is still in the list result.
- During a condition query, even if the marker does not exist in the list actually, what is returned is printed starting from the next to what conforms to the marker letter sorting. If the max-keys value is less than 0 or greater than 1000, error 400 Bad Request is returned. The error code is “InvalidArgument” .
- If the prefix, marker, or delimiter parameters do not meet the length requirement, 400 Bad Request is returned. The error code is “InvalidArgument” .
- The prefix and marker parameters are used to achieve display by pages, and the parameter length must be less than 1024 bytes.
- Setting a prefix as the name of a folder lists the files starting with this prefix, recursively returning all files and subfolders in this folder. Additionally, if we

set the Delimiter as “/”, the returned values lists the files in the folder and the subfolders are returned in the CommonPrefixes section. Recursive files and folders in the subfolders are not displayed. For example, a bucket has the following three objects: fun/test.jpg, fun/movie/001.avi, and fun/movie/007.avi. If the prefix is set to “fun/”, three objects are returned. If the delimiter is set to “/” additionally, file “fun/test.jpg” and prefix “fun/movie/” are returned. That is, the folder logic is achieved.

### Scenario example

Four objects are available in the bucket “my\_oss” and are named as:

- oss.jpg
- fun/test.jpg
- fun/movie/001.avi
- fun/movie/007.avi

### Example

#### Request example:

```
GET / HTTP / 1 . 1
Host : oss - example . oss - cn - hangzhou . aliyuncs . com
Date : Fri , 24 Feb 2012 08 : 43 : 27 GMT
Authorizat ion : OSS qn6qrrqxo2 oawuk53otf jbyc : BC + oQIXVR2
/ ZghT7cGa0y kbo04M =
```

#### Return example:

```
HTTP / 1 . 1 200 OK
x - oss - request - id : 534B371674 E88A4D8906 008B
Date : Fri , 24 Feb 2012 08 : 43 : 27 GMT
Content - Type : applicatio n / xml
Content - Length : 1866
Connection : keep - alive
Server : AliyunOSS
<? xml version =" 1 . 0 " encoding =" UTF - 8 "? >
< ListBucket Result xmlns =" http :// doc . oss - cn - hangzhou .
aliyuncs . com ">
< Name > oss - example </ Name >
< Prefix ></ Prefix >
< Marker ></ Marker >
< MaxKeys > 100 </ MaxKeys >
< Delimiter ></ Delimiter >
  < IsTruncate d > false </ IsTruncate d >
  < Contents >
    < Key > fun / movie / 001 . avi </ Key >
    < LastModifi ed > 2012 - 02 - 24T08 : 43 : 07 . 000Z </
LastModifi ed >
    < ETag >& quot ; 5B3C1A2E05 3D763E1B00 2CC607C5A0 FE & quot
;</ ETag >
    < Type > Normal </ Type >
    < Size > 344606 </ Size >
```

```

    < StorageClass > Standard </ StorageClass >
    < Owner >
      < ID > 0022012022_2 </ ID >
      < DisplayName > user - example </ DisplayName >
    </ Owner >
  </ Contents >
  < Contents >
    < Key > fun / movie / 007 . avi </ Key >
    < LastModified > 2012 - 02 - 24T08 : 43 : 27 . 000Z </
LastModified >
    < ETag > " 5B3C1A2E05_3D763E1B00_2CC607C5A0_FE "
; </ ETag >
    < Type > Normal </ Type >
    < Size > 344606 </ Size >
    < StorageClass > Standard </ StorageClass >
    < Owner >
      < ID > 0022012022_2 </ ID >
      < DisplayName > user - example </ DisplayName >
    </ Owner >
  </ Contents >
  < Contents >
    < Key > fun / test . jpg </ Key >
    < LastModified > 2012 - 02 - 24T08 : 42 : 32 . 000Z </
LastModified >
    < ETag > " 5B3C1A2E05_3D763E1B00_2CC607C5A0_FE "
; </ ETag >
    < Type > Normal </ Type >
    < Size > 344606 </ Size >
    < StorageClass > Standard </ StorageClass >
    < Owner >
      < ID > 0022012022_2 </ ID >
      < DisplayName > user - example </ DisplayName >
    </ Owner >
  </ Contents >
  < Contents >
    < Key > oss . jpg </ Key >
    < LastModified > 2012 - 02 - 24T06 : 07 : 48 . 000Z </
LastModified >
    < ETag > " 5B3C1A2E05_3D763E1B00_2CC607C5A0_FE "
; </ ETag >
    < Type > Normal </ Type >
    < Size > 344606 </ Size >
    < StorageClass > Standard </ StorageClass >
    < Owner >
      < ID > 0022012022_2 </ ID >
      < DisplayName > user - example </ DisplayName >
    </ Owner >
  </ Contents >
</ ListBucketResult >

```

### Example of a request containing the prefix parameter:

```

GET /? prefix = fun HTTP / 1 . 1
Host : oss - example . oss - cn - hangzhou . aliyuncs . com
Date : Fri , 24 Feb 2012 08 : 43 : 27 GMT
Authorization : OSS qn6qrrqxo2 oawuk53otf jbyc : BC + oQIXVR2
/ ZghT7cGa0y kbo04M =

```

### Return example:

```

HTTP / 1 . 1 200 OK
x - oss - request - id : 534B371674 E88A4D8906 008B

```

```

Date : Fri , 24 Feb 2012 08 : 43 : 27 GMT
Content - Type : applicatio n / xml
Content - Length : 1464
Connection : keep - alive
Server : AliyunOSS
<? xml version =" 1 . 0 " encoding =" UTF - 8 "? >
< ListBucket Result xmlns =" http :// doc . oss - cn - hangzhou .
aliyuncs . com ">
< Name > oss - example </ Name >
< Prefix > fun </ Prefix >
< Marker ></ Marker >
< MaxKeys > 100 </ MaxKeys >
< Delimiter ></ Delimiter >
  < IsTruncate d > false </ IsTruncate d >
  < Contents >
    < Key > fun / movie / 001 . avi </ Key >
    < LastModifi ed > 2012 - 02 - 24T08 : 43 : 07 . 000Z </
LastModifi ed >
    < ETag >& quot ; 5B3C1A2E05 3D763E1B00 2CC607C5A0 FE & quot
; </ ETag >
    < Type > Normal </ Type >
    < Size > 344606 </ Size >
    < StorageCla ss > Standard </ StorageCla ss >
    < Owner >
      < ID > 0022012022 2 </ ID >
      < DisplayNam e > user_examp le </ DisplayNam e >
    </ Owner >
  </ Contents >
  < Contents >
    < Key > fun / movie / 007 . avi </ Key >
    < LastModifi ed > 2012 - 02 - 24T08 : 43 : 27 . 000Z </
LastModifi ed >
    < ETag >& quot ; 5B3C1A2E05 3D763E1B00 2CC607C5A0 FE & quot
; </ ETag >
    < Type > Normal </ Type >
    < Size > 344606 </ Size >
    < StorageCla ss > Standard </ StorageCla ss >
    < Owner >
      < ID > 0022012022 2 </ ID >
      < DisplayNam e > user_examp le </ DisplayNam e >
    </ Owner >
  </ Contents >
  < Contents >
    < Key > fun / test . jpg </ Key >
    < LastModifi ed > 2012 - 02 - 24T08 : 42 : 32 . 000Z </
LastModifi ed >
    < ETag >& quot ; 5B3C1A2E05 3D763E1B00 2CC607C5A0 FE & quot
; </ ETag >
    < Type > Normal </ Type >
    < Size > 344606 </ Size >
    < StorageCla ss > Standard </ StorageCla ss >
    < Owner >
      < ID > 0022012022 2 </ ID >
      < DisplayNam e > user_examp le </ DisplayNam e >
    </ Owner >
  </ Contents >
</ ListBucket Result >

```

**Example of a request containing parameters prefix and delimiter:**

```

GET /? prefix = fun /& delimiter =/ HTTP / 1 . 1
Host : oss - example . oss - cn - hangzhou . aliyuncs . com
Date : Fri , 24 Feb 2012 08 : 43 : 27 GMT

```



```
Authorization : OSS qn6qrrqxo2 oawuk53otf jbyc : DNrn7xHk3
sgysx7I8U9 I9IY1vY =
```

### Return example:

```
HTTP / 1 . 1 200 OK
x - oss - request - id : 534B371674 E88A4D8906 008B
Date : Fri , 24 Feb 2012 08 : 43 : 27 GMT
Content - Type : application / xml
Content - Length : 712
Connection : keep - alive
Server : AliyunOSS
<? xml version =" 1 . 0 " encoding =" UTF - 8 "? >
< ListBucket Result xmlns =" http :// doc . oss - cn - hangzhou .
aliyuncs . com ">
< Name > oss - example </ Name >
< Prefix > fun </ Prefix >
< Marker ></ Marker >
< MaxKeys > 100 </ MaxKeys >
< Delimiter ></ Delimiter >
  < IsTruncate d > false </ IsTruncate d >
  < Contents >
    < Key > fun / test . jpg </ Key >
    < LastModifi ed > 2012 - 02 - 24T08 : 42 : 32 . 000Z </
LastModifi ed >
    < ETag >& quot ; 5B3C1A2E05 3D763E1B00 2CC607C5A0 FE & quot
; </ ETag >
    < Type > Normal </ Type >
    < Size > 344606 </ Size >
    < StorageCla ss > Standard </ StorageCla ss >
    < Owner >
      < ID > 0022012022 2 </ ID >
      < DisplayNam e > user_examp le </ DisplayNam e >
    </ Owner >
  </ Contents >
  < CommonPref ixes >
    < Prefix > fun / movie </ Prefix >
  </ CommonPref ixes >
</ ListBucket Result >
```

## 3.5 GetBucketInfo

GetBucketInfo operation is used to view the bucket information.

The information includes the following:

- Create time
- Internet access endpoint
- Intranet access endpoint
- Bucket owner information
- Bucket ACL (AccessControlList)

### Request syntax

```
GET /? bucketInfo HTTP / 1 . 1
```

```
Host : BucketName . oss . aliyuncs . com
Date : GMT Date
Authorization : SignatureV alue
```

## Response elements

| Name              | Type      | Description   |
|-------------------|-----------|---|
| BucketInfo        | Container | The container that saves the bucket information content<br>Sub-node: Bucket node<br>Parent node: none           |
| Bucket            | Container | The container that saves the bucket specific information<br>Parent node: BucketInfo node                        |
| CreationDate      | time      | The creation time of the bucket. Time format: 2013-07-31T10:56:21.000Z<br>Parent node: BucketInfo.Bucket        |
| ExtranetEndpoint  | string    | The Internet domain name that the bucket accesses<br>Parent node: BucketInfo.Bucket                             |
| IntranetEndpoint  | string    | The intranet domain name for accessing the bucket from ECS in the same region<br>Parent node: BucketInfo.Bucket |
| Location          | string    | The region of the data center that the bucket is located in<br>Parent node: BucketInfo.Bucket                   |
| Name              | string    | The bucket name<br>Parent node: BucketInfo.Bucket   |
| Owner             | container | Container used for saving the information about the bucket owner.<br>Parent node: BucketInfo.Bucket             |
| ID                | string    | User ID of the bucket owner.<br>Parent node: BucketInfo.Bucket.Owner  |
| DisplayName       | string    | Name of the bucket owner (the same as the ID currently).<br>Parent node: BucketInfo.Bucket.Owner                |
| AccessControlList | container | Container used for storing the ACL information<br>Parent node: BucketInfo.Bucket                                |

| Name               | Type               | Description   |
|--------------------|--------------------|---|
| Grant              | enumerative string | ACL permissions of the bucket.<br>Valid values: private , public - read , and public - read - write<br>Parent node: BucketInfo.Bucket.AccessControlList |
| DataRedundancyType | enumerative string | The data redundancy type of the bucket.<br>Valid values: LRS and ZRS<br>Parent node: BucketInfo.Bucket  |

### Detail analysis

- If the bucket does not exist, error 404 is returned. Error code: NoSuchBucket.
- Only the owner of a bucket can view the information of the bucket. If other users attempt to access the location information, the error 403 Forbidden with the error code: AccessDenied is returned.
- The request can be initiated from any OSS endpoint.

### Example

#### Request example:

```
Get /? bucketInfo HTTP / 1 . 1
Host : oss - example . oss . aliyuncs . com
Date : Sat , 12 Sep 2015 07 : 51 : 28 GMT
Authorization : OSS qn6qrrqxo2 oawuk53otf jbyc : BuG4rRK +
zNhH1AcF51 NNHD39zXw =
```

#### Return example after the bucket information is obtained successfully:

```
HTTP / 1 . 1 200
x - oss - request - id : 534B371674 E88A4D8906 008B
Date : Sat , 12 Sep 2015 07 : 51 : 28 GMT
Connection : keep - alive
Content - Length : 531
Server : AliyunOSS

<? xml version =" 1 . 0 " encoding =" UTF - 8 "? >
< BucketInfo >
  < Bucket >
    < CreationDate > 2013 - 07 - 31T10 : 56 : 21 . 000Z </
CreationDate >
    < ExtranetEndpoint > oss - cn - hangzhou . aliyuncs . com </
ExtranetEndpoint >
    < IntranetEndpoint > oss - cn - hangzhou - internal . aliyuncs .
com </ IntranetEndpoint >
    < Location > oss - cn - hangzhou </ Location >
    < Name > oss - example </ Name >
    < Owner >
      < DisplayName > username </ DisplayName >
      < ID > 2718347391 43143 </ ID >
```

```

    </ Owner >
    < AccessCont rolList >
      < Grant > private </ Grant >
    </ AccessCont rolList >
  </ Bucket >
</ BucketInfo >

```

**Return example if the requested bucket information does not exist:**

```

HTTP / 1 . 1 404
x - oss - request - id : 534B371674 E88A4D8906 009B
Date : Sat , 12 Sep 2015 07 : 51 : 28 GMT
Connection : keep - alive
Content - Length : 308
Server : AliyunOSS

<? xml version =" 1 . 0 " encoding =" UTF - 8 "? >
< Error >
  < Code > NoSuchBuck et </ Code >
  < Message > The specified bucket does not exist .</
Message >
  < RequestId > 568D547F31 243C673BA1 4274 </ RequestId >
  < HostId > nosuchbuck et . oss . aliyuncs . com </ HostId >
  < BucketName > nosuchbuck et </ BucketName >
</ Error >

```

**Return example if the requester has no access permission to the bucket information:**

```

HTTP / 1 . 1 403
x - oss - request - id : 534B371674 E88A4D8906 008C
Date : Sat , 12 Sep 2015 07 : 51 : 28 GMT
Connection : keep - alive
Content - Length : 209
Server : AliyunOSS

<? xml version =" 1 . 0 " encoding =" UTF - 8 "? >
< Error >
  < Code > AccessDeni ed </ Code >
  < Message > AccessDeni ed </ Message >
  < RequestId > 568D5566F2 D0F89F5C0E B66E </ RequestId >
  < Hostid > test . oss . aliyuncs . com </ hostid >
</ Error >

```

## 3.6 DeleteBucketWebsite

DeleteBucketWebsite is used to disable the static website hosting mode and the redirection rules for a bucket. Only the owner of a bucket can disable the static website hosting mode for the bucket.

**Request syntax**

```

DELETE /? website HTTP / 1 . 1
Host : BucketName . oss - cn - hangzhou . aliyuncs . com
Date : GMT Date

```

```
Authorization : SignatureV alue
```

## Examples

### Request example

```
DELETE /? website HTTP / 1 . 1
Host : BucketName . oss - cn - hangzhou . aliyuncs . com
Date : Fri , 24 Feb 2012 05 : 45 : 34 GMT
Authorization : OSS qn6qrrqxo2 oawuk53otf jbyc : LnM4AZ10eI
duZF5vGFWi cOMEkVg =
```

### Response example

```
HTTP / 1 . 1 204 No Content
x - oss - request - id : 534B371674 E88A4D8906 008B
Date : Fri , 24 Feb 2012 05 : 45 : 34 GMT
Connection : keep - alive
Content - Length : 0
Server : AliyunOSS
```

### Complete code

```
DELETE /? website HTTP / 1 . 1
Date : Fri , 27 Jul 2018 09 : 10 : 52 GMT
Host : test . oss - cn - hangzhou - internal . aliyuncs . com
Authorization : OSS alnBNgkzzx cQMf8u : qPrKwuMaar A4Tfk1pqTC
ylFs1jY =
User - Agent : aliyun - sdk - python - test / 0 . 4 . 0

HTTP / 1 . 1 204 No Content
Server : AliyunOSS
Date : Fri , 27 Jul 2018 09 : 10 : 52 GMT
Content - Length : 0
Connection : keep - alive
x - oss - request - id : 5B5AE19C18 8DC1CE81DA D7C8
```

## Error code

| Error code   | HTTP status code | Description  |
|--------------|------------------|--|
| NoSuchBucket | 404 Not Found    | The bucket that you want to delete does not exist.   |
| AccessDenied | 403 Forbidden    | You do not have the permission to disable the static website hosting mode for the bucket. Only the owner of a bucket can disable the static website hosting mode for a bucket. |

## 3.7 PutBucketLifecycle

The bucket owner can set the lifecycle of a bucket with the `PutBucketLifecycle` request. After Lifecycle is enabled, OSS automatically deletes the objects or transitions the objects (to another storage class) corresponding the lifecycle rules on a regular basis.

### Request syntax

```
PUT /? lifecycle HTTP / 1 . 1
Date : GMT Date
Content - Length : ContentLength
Content - Type : application/xml
Authorization : SignatureValue
Host : BucketName.oss.aliyuncs.com
<? xml version =" 1 . 0 " encoding =" UTF - 8 ">
< LifecycleConfiguration >
  < Rule >
    < ID > RuleID </ ID >
    < Prefix > Prefix </ Prefix >
    < Status > Status </ Status >
    < Expiration >
      < Days > Days </ Days >
    </ Expiration >
    < AbortMulti partUpload >
      < Days > Days </ Days >
    </ AbortMulti partUpload >
  </ Rule >
</ LifecycleConfiguration >
```

### Request elements

| Name              | Type   | Required?                                    | Description  |
|-------------------|--------|--|--|
| CreatedBeforeDate | string | One from the two: Days and CreatedBeforeDate | Specify the time before which the rules go into effect. The date must conform to the ISO8601 format and always be UTC 00:00. For example: 2002-10-11T00:00:00.000Z, which means the objects with a last update time before 2002-10-11T00:00:00.000Z are deleted or transitioned to another storage class, and the objects updated after this time (including this time) are not deleted or transitioned.<br>Parent node: Expiration or AbortMulti partUpload |

| Name                   | Type             | Required?                                    | Description  |
|------------------------|------------------|--|--|
| Days                   | positive integer | One from the two: Days and CreatedBeforeDate | Specify how many days after the last object update until the rules take effect.<br>Parent node: Expiration   |
| Expiration             | container        | No   | Specify the expiration attribute of the object.<br>Sub-node: Days or CreatedBeforeDate<br>Parent node: Rule  |
| AbortMulti partUpload  | container        | No   | Specify the expiration attribute of the unfulfilled Part rules.<br>Sub-node: Days or CreatedBeforeDate<br>Parent node: Rule  |
| ID                     | string           | No   | The unique ID of a rule. An ID is composed of 255 bytes at most. When you fail to specify this value or this value is null, OSS generates a unique value for you.<br>Sub-node: none<br>Parent node: Rule |
| LifecycleConfiguration | container        | Yes  | Container used for storing lifecycle configurations, which can hold a maximum of 1000 rules.<br>Sub-node: Rule<br>Parent node: none  |
| Prefix                 | string           | Yes  | Specify the prefix applicable to a rule. Only those objects with a matching prefix can be affected by the rule. It cannot be overlapped.<br>Sub-node: none<br>Parent node: Rule                          |
| Rule                   | container        | Yes  | Express a rule<br>Sub-nodes: ID, Prefix, Status, Expiration<br>Parent node: LifecycleConfiguration   |
| Status                 | string           | Yes  | If this value is Enabled, OSS runs this rule regularly. If this value is Disabled, then OSS ignores this rule.<br>Parent node: Rule<br>Valid value: Enabled , Disabled                                   |

| Name         | Type      | Required?                                 | Description  |
|--------------|-----------|---|--|
| StorageClass | string    | Required if parent node transition is set | Specifies the type of target storage that the object is transition to the OSS.<br>Value: IA , Archive<br>Parent node: Transition |
| Transition   | Container | No  | Specifies when the object is transition to the IA or archive storage type during a valid life cycle.                             |

### Detail analysis

- Only the bucket owner can initiate a Put Bucket Lifecycle request. Otherwise, the message of 403 Forbidden is returned. Error code: AccessDenied.
- If no lifecycle has been set previously, this operation creates a new lifecycle configuration or overwrites the previous configuration.
- You can also set an expiration time for an object, or for the Part. Here, the Part refers to the unsubmitted parts for multipart upload.

### Notes for storage types transition:

- Supports objects in Standard bucket transition to IA and Archive storage type. Standard bucket can simultaneously configure both transition to IA and archive storage type rules for one object. In this case, the time set to transition to archive must be longer than the time to transition to IA. For example, the days set for transition to IA is 30, then it must be greater than 30 days set for transition to archive. Otherwise, the invalidargument error is returned.
- The object setting must have an expiration time greater than the time converted to IA or archive. Otherwise, the invalidArgument error is returned.
- Supports objects transition to archive storage type in IA bucket.
- Archive bucket creation is not supported.
- IA object conversion is not supported as standard.
- The archive object conversion is not supported for IA or standard.

### Examples

#### Request example:

```
PUT /? lifecycle HTTP / 1 . 1
Host : oss - example . oss . aliyuncs . com
Content - Length : 443
Date : Mon , 14 Apr 2014 01 : 08 : 38 GMT
```



```

Authorization : OSS qn6qrrqxo2 oawuk53otf jbyc : KU5h8YMUC7
8M30dXqf3J xrTzHiA =
<? xml version =" 1 . 0 " encoding =" UTF - 8 ">
</ LifecycleC onfigurati on >
  < Rule >
    < ID > delete objects and parts after one day </ ID >
    < Prefix > logs </ Prefix >
    < Status > Enabled </ Status >
    < Expiration >
      < Days > 1 </ Days >
    </ Expiration >
    < AbortMulti partUpload >
      < Days > 1 </ Days >
    </ AbortMulti partUpload >
  </ Rule >
  < Rule >
    < ID > delete created before date </ ID >
    < Prefix > backup </ Prefix >
    < Status > Enabled </ Status >
    < Expiration >
      < CreatedBef oreDate > 2014 - 10 - 11T00 : 00 : 00 . 000Z </
CreatedBef oreDate >
    </ Expiration >
    < AbortMulti partUpload >
      < CreatedBef oreDate > 2014 - 10 - 11T00 : 00 : 00 . 000Z </
CreatedBef oreDate >
    </ AbortMulti partUpload >
  </ Rule >
</ LifecycleC onfigurati on >

```

**Response example:**

```

HTTP / 1 . 1 200 OK
x - oss - request - id : 534B371674 E88A4D8906 008B
Date : Thu , 8 Jun 2017 13 : 08 : 38 GMT
Content - Length : 0
Connection : keep - alive
Server : AliyunOSS

```

## 4 Object operations

---

### 4.1 PutObject


PutObject is used to upload files.


#### Request syntax


```
PUT / ObjectName HTTP / 1 . 1
Content - Length : ContentLength
Content - Type : ContentType
Host : BucketName . oss - cn - hangzhou . aliyuncs . com
Date : GMT Date
Authorization : SignatureValue
```

#### Request header

| Parameter             | Type   | Description   |
|-----------------------|--------|---|
| Authorization         | String | Indicates that the request is authorized. For more information, see <a href="#">RFC2616</a> .<br>Default value: none                              |
| Cache - control       | String | Specifies the Web page caching behavior when the object is downloaded. For more information, see <a href="#">RFC2616</a> .<br>Default value: none |
| Content - Disposition | String | Specifies the name of the object when the object is downloaded. For more information, see <a href="#">RFC2616</a> .<br>Default value: none        |
| Content - Encoding    | String | Specifies the content encoding format when the object is downloaded. For more information, see <a href="#">RFC2616</a> .<br>Default value: none   |

| Parameter        | Type   | Description  |
|------------------|--------|--|
| Content - Md5    | String | <p>As defined in RFC 1864, an MD5 value, which is a 128-bit number, is calculated based on the message content, rather than the header. This number is then Base64-encoded into a Content-MD5 value. This request header can be used to check the validity of a message. More specifically, the name can be used whether the message content is consistent with the sent content. Although this request header is optional, we recommend that you use this request header for end-to-end checks.</p> <p>Default value: none</p> <p>Restriction: none</p> |
| Content - Length | String | <p>If the value of Content - Length in the request header is smaller than the data length in the request body, OSS can still create the object successfully. However, the object size is the value of Content - Length , and the data that exceeds the value is discarded.</p>   |
| ETag             | String | <p>An entity tag (ETag) is created to identify the content of an object when the object is created. For an object created with the PutObject request, its ETag is the MD5 value of the object content. For an object created by using other methods, its ETag is the UUID of the object content. The ETag value of an object can be used to check whether the object content has changed. However, we recommend that you not use the ETag of an object as the MD5 value of the object to verify data integrity.</p> <p>Default value: None</p>           |
| Expires          | String | <p>Specify the expiration time. For more information, see <a href="#">RFC2616</a>.</p> <p>Default value: none</p> <div>  <p>Note:<br/>OSS does not limit and verify this value.</p> </div>  |

| Parameter  | Type   | Description  |
|--|--------|--|
| <code>x-oss-server-side-encryption</code>        | String | <p>Specifies the server-side encryption algorithm when OSS creates an object.<br/>Valid value: AES256 or KMS</p> <div>  <b>Note:</b><br/>           You must enable KMS (Key Management Service) in the console before you can use the KMS encryption algorithm. Otherwise, a KmsServiceNotEnabled error code is reported.         </div> |
| <code>x-oss-server-side-encryption-key-id</code> | String | <p>Specifies the primary key managed by KMS.<br/>This parameter is valid when the value of <code>x-oss-server-side-encryption</code> is set to KMS.</p>  |
| <code>x-oss-object-acl</code>                    | String | <p>Specifies the access permission when OSS creates an object.<br/>Valid values: <code>public-read</code>, <code>private</code>, and <code>public-read-write</code></p>  |

| Parameter                 | Type   | Description  |
|---------------------------|--------|--|
| x - oss - storage - class | String | <p>Specifies the storage class of the object.</p> <p>Valid values:</p> <ul style="list-style-type: none"> <li>Standard</li> <li>IA</li> <li>Archive</li> </ul> <p>Supported interfaces: PutObject, InitMultiPartUpload, AppendObject, PutObjectSymlink, and CopyObject</p> <div>  <p><b>Note:</b></p> <ul style="list-style-type: none"> <li>The status code of 400 Bad Request is returned if the value of StorageClass is invalid. Error description: InvalidArgument.</li> <li>If you specify the value of x-oss-storage-class when uploading an object to a bucket, the storage class of the uploaded object is the specified value of x-oss-storage-class. For example, if you specify the value of x-oss-storage-class to Standard when uploading an object to a bucket of the IA storage class, the storage class of the object is Standard.</li> </ul> </div> |

#### Detail analysis

- For the object that you want to upload:
  - A status code 403 Forbidden error is returned if you do not have access to the bucket. Error description: AccessDenied.
  - If an object with the same name already exists and you have access to it, the existing object is overwritten by the uploaded object, and the status code 200 OK is returned.
  - The status code 404 Not Found is returned if the bucket does not exist. Error description: NoSuchBucket.
- The status code 400 Bad Request is returned if the length of the input object key exceeds 1023 bytes. Error description: InvalidObjectName.
- Content-Length
  - If the value of Content - Length in the request header is smaller than the data length in the request body, OSS can still create the object successfully.

However, the object size is the value of `Content - Length` , and the data that exceeds the value is discarded.

- The status code 400 Bad request is returned if the length of the uploaded object exceeds 5 GB. Error description: `InvalidArgument`.
- If the length is set, but the message body is not sent or the size of the sent body is smaller than the specified size, the server waits until timeout, and the status code 400 Bad Request is returned. Error description: `RequestTimeout`.
- HTTP header
  - OSS supports the following five header fields defined in HTTP: `Cache-Control`, `Expires`, `Content-Encoding`, `Content-Disposition`, and `Content-Type`. If these headers are set when you upload an object, the header values are automatically set to the values set in the upload when the object is downloaded. For more information about the header fields, see [RFC2616](#).
  - If the header is not encoded in the [chunked encoding](#) method and the `Content-Length` parameter is not added, a 411 Length Required error is returned. Error description: `MissingContentLength`.
  - If you specify the `x-oss-server-side-encryption` header when you perform a `PutObject` operation, the value of this header must be set to `AES256`. Otherwise , the status code 400 Bad Request is returned. Error description: `InvalidEncryptionAlgorithmError`. After this header is specified, it is also included in the response header, and OSS encrypts the uploaded object by using the specified method. When this object is downloaded, `x-oss-server-side-encryption` is included in the response header, and the value of `x-oss-server-side-encryption` is set to the encryption algorithm of this object.
  - If the `PutObject` request carries a parameter with the `x-oss-meta-` prefix, the parameter is considered as user meta, for example, `x-oss-meta-location`. A single object can have multiple parameters with the `x-oss-meta-` prefix. However , the total size of all user meta cannot exceed 8 KB. User meta can contain alphanumeric characters, en dashes (–), spaces ( ), and double quotation marks ("). Other characters including underscores (–) are not supported.

#### Content-MD5 calculation error

According to the standard, the Content-MD5 value is calculated as follows: Calculate the MD5-encrypted 128-bit binary array, and then encode the binary array (but not the 32-bit string) with Base64.

For example, if the content you want to upload is `0123456789`, the Content-MD5 value of the string can be calculated as follows:

The correct calculation method can be implemented in Python as follows:

```
>>> import base64, hashlib
>>> hash = hashlib.md5()
>>> hash.update("0123456789")
>>> base64.b64encode(hash.digest())
' eB5eJF1ptW aXm4bijSPy xw =='
```



#### Note:

**Correct calculation method:** Use `hash.digest()` to calculate the 128-bit binary array first. For example: `>>> hash.digest() ' x \ x1e ^$] i \ xb5f \ x97 \ x9b \ x86 \ xe2 \ x8d #\ xf2 \ xc7 '`

A common error in calculations is to encode the calculated 32-bit string with Base64 to obtain the Content-MD5 value. For example, `hash.hexdigest()` is used to calculate a visible 32-bit string. `>>> hash.hexdigest() ' 781e5e245d 69b566979b 86e28d23f2 c7 '` If you encode the incorrect MD5 value with Base64, the result is as follows. `>>> base64.b64encode(hash.hexdigest()) ' NzgxZTVlMj Q1ZDY5YjU2 Njk3OWI4Nm UyOGQyM2Yy Yzc ='`

## Examples

### · Example 1

Request example in a simple upload:

```
PUT / test . txt HTTP / 1 . 1
Host : test . oss - cn - zhangjiako u . aliyuncs . com
User - Agent : aliyun - sdk - python / 2 . 6 . 0 ( Windows / 7 /
AMD64 ; 3 . 7 . 0 )
Accept : */*
Connection : keep - alive
Content - Type : text / plain
date : Tue , 04 Dec 2018 15 : 56 : 37 GMT
authorizat ion : OSS qn6qrrqxo2 oawuk53otf jbyc : kZoYNv66bs
mc10 + dcGKw5x2PR rk =
Transfer - Encoding : chunked
```

Response example:

```
HTTP / 1 . 1 200 OK
Server : AliyunOSS
Date : Tue , 04 Dec 2018 15 : 56 : 38 GMT
Content - Length : 0
Connection : keep - alive
```

```
x - oss - request - id : 5C06A3B67B 8B5A3DA422 299D
ETag : " D41D8CD98F 00B204E980 0998ECF842 7E "
x - oss - hash - crc64ecma : 0
Content - MD5 : 1B2M2Y8Asg TpgAmY7PhC fg ==
x - oss - server - time : 7
```

• **Example 2**

**Request example in which the storage class is Archive:**

```
PUT / oss . jpg HTTP / 1 . 1
Host : oss - example . oss - cn - hangzhou . aliyuncs . com
Cache - control : no - cache
Expires : Fri , 28 Feb 2012 05 : 38 : 42 GMT
Content - Encoding : utf - 8
Content - Dispositio n : attachment ; filename = oss_downlo ad .
jpg
Date : Fri , 24 Feb 2012 06 : 03 : 28 GMT
Content - Type : image / jpg
Content - Length : 344606
x - oss - storage - class : Archive
Authorizat ion : OSS qn6qrrqxo2 oawuk53otf jbyc : kZoYNv66bs
mc10 + dcGKw5x2PR rk =

[ 344606 bytes of object data ]
```

**Response example:**

```
HTTP / 1 . 1 200 OK
Server : AliyunOSS
Date : Sat , 21 Nov 2015 18 : 52 : 34 GMT
Content - Type : image / jpg
Content - Length : 0
Connection : keep - alive
x - oss - request - id : 5650BD7220 7FB3044396 2F9A
x - oss - bucket - version : 1418321259

ETag : " A797938C31 D59EDD08D8 6188F6D5B8 72 "
```



**Note:**

For more example code for PutObject, see [Upload objects](#).

## 4.2 CopyObject

CopyObject is used to copy objects within a bucket or between buckets in the same region.

The CopyObject interface sends a PUT request to OSS and adds the `x - oss -`

`copy - source` element in the PUT request header to specify the source object.

OSS recognizes the request as a copy operation and perform this operation on the server. If the copy operation is successful, the information about the target object is returned.





## Request syntax

```
PUT / DestObject Name HTTP / 1 . 1
Host : DestBucket Name . oss - cn - hangzhou . aliyuncs . com
Date : GMT Date
Authorization : SignatureV alue
x - oss - copy - source : / SourceBucketName / SourceObjectName
```


## Request header

| Name  | Type   | Description  |
|---|--------|--|
| x - oss - copy - source                           | String | Specifies the address of the source object.<br>Default value: None.  |
| x - oss - copy - source - if - match              | String | If the ETag of the source object is the same as the ETag provided by the user, the copy operation is performed and a 200 OK message is returned. Otherwise, a 412 HTTP error code (preprocessing failed) is returned.<br>Default value: None.                          |
| x - oss - copy - source - if - none - match       | String | If the ETag of the source object is different from the ETag provided by the user, the copy operation is performed and a 200 OK message is returned. Otherwise, a 304 HTTP error code (preprocessing failed) is returned.<br>Default value: None.                       |
| x - oss - copy - source - if - unmodified - since | String | If the time specified in the received parameter is the same as or later than the modification time of the object, the object is copied normally and a 200 OK message is returned. Otherwise, a 412 Precondition Failed error code is returned.<br>Default value: None. |
| x - oss - copy - source - if - modified - since   | String | If the source object is modified after the time specified by the user, the copy operation is performed. Otherwise, a 304 HTTP error code (preprocessing failed) is returned.<br>Default value: None.   |

| Name                                  | Type   | Description  |
|---------------------------------------|--------|--|
| <code>x-oss-metadata-directive</code> | String | <p>Specifies how to set the metadata of the target object. The valid values are COPY and REPLACE.</p> <ul style="list-style-type: none"> <li>• <b>Copy (default):</b> The metadata of the source object is copied to the target object. The <code>x-oss-server-side-encryption</code> of the source object is not copied. That is, server-side encryption is performed on the target object only if the <code>x-oss-server-side-encryption</code> header is specified in the COPY request.</li> <li>• <b>REPLACE :</b> The metadata of the target object is set to the metadata specified in the user's request instead of the metadata of the source object.</li> </ul> <div>  <b>Note:</b><br/>           If the source object and the target object have the same address, the metadata of the target object is replaced with the metadata of the source object regardless of the value of <code>x-oss-metadata-directive</code> .         </div> |

| Name  | Type   | Description   |
|---|--------|---|
| x - oss - server - side - encryption            | String | <p>Specifies the server-side entropy encoding encryption algorithm when OSS creates the target object.</p> <p>Valid values:</p> <ul style="list-style-type: none"> <li>• AES256</li> <li>• KMS (You must enable KMS in the console before you can use the KMS encryption algorithm. Otherwise, a KmsServiceNotEnabled error code is returned.)</li> </ul> <div>  <b>Note:</b> <ul style="list-style-type: none"> <li>• If the x - oss - server - side - encryption header is not specified in the copy operation, the target object is not encrypted on the server side no matter whether server-side encryption has been performed on the source object.</li> <li>• If you specify the x - oss - server - side - encryption header, server-side encryption is performed on the target object no matter whether the encryption has been performed on the source object. In addition, the response header for the copy request includes the x - oss - server - side - encryption header, and the value of the header is the encryption algorithm of the target object. When the target object is downloaded, the response header also includes the x - oss - server - side - encryption header, and the value of the header is the encryption algorithm of the target object.</li> </ul> </div> |
| x - oss - server - side - encryption - key - id | String | <p>Indicates the primary key managed by KMS. This parameter is valid when the value of x - oss - server - side - encryption is KMS.</p>   |

| Name                      | Type   | Description  |
|---------------------------|--------|--|
| x - oss -<br>object - acl | String | <b>Specifies the ACL for the target object when it is created.</b><br><b>Valid values:</b> <ul style="list-style-type: none"><li>• public - read</li><li>• private</li><li>• public - read - write</li></ul> |

| Name                         | Type   | Description   |
|------------------------------|--------|---|
| x - oss -<br>storage - class | String | <p>Specifies the storage class of the object.<br/>Valid values:</p> <ul style="list-style-type: none"> <li>Standard</li> <li>IA</li> <li>Archive</li> </ul> <p>Supported interfaces: PutObject, InitMultiPartUpload, AppendObject, PutObjectSymlink, and CopyObject</p> <div>  <p><b>Note:</b></p> <ul style="list-style-type: none"> <li>If the value of StorageClass is invalid, a 400 error message is returned with an error code: InvalidArgument.</li> <li>We recommend that you do not set the storage class in PutObjectSymlink to IA or Archive because an IA or Archive object smaller than 64 KB is billed at 64 KB.</li> <li>If you specify the value of x-oss-storage-class when uploading an object to a bucket, the storage class of the uploaded object is the specified value of x-oss-storage-class. For example, if you specify the value of x-oss-storage-class to Standard when uploading an object to a bucket of the IA storage class, the storage class of the object is Standard.</li> <li>If you change the storage class of an object, the object is considered as overwritten and will incur charges. An object of the IA or Archive class will be charged if it is overwritten within 30 and 60 days respectively after it is created. For example, if you change the storage class of an object from IA to Archive or Standard 10 days after the object is created, early deletion fees for 20 days will be charged.</li> </ul> </div> |

## Response elements

Table 4-1: Response elements

| Name             | Type   | Description   |
|------------------|--------|---|
| CopyObjectResult | String | Indicates the result of CopyObject.<br>Default value: None.                                     |
| ETag             | String | Indicates the ETag of the target object.<br>Parent element: CopyObjectResult                    |
| LastModified     | String | Indicates the time when the target object is last modified.<br>Parent element: CopyObjectResult |

## Detail analysis

- Limits
  - CopyObject only supports objects smaller than 1 GB. To copy objects larger than 1 GB, you must use MultipartUpload. For more information, see [UploadPartCopy](#).
  - The requester must have the read permission on the source object.
  - The source object and the target object must be in the same region.
  - You cannot copy objects created with AppendObject.
  - If the source object is a symbolic link, only the symbolic link (instead of the content that the link directs to) is copied.
- Billing items
  - A GET request is billed according to the bucket where the source object is stored.
  - A PUT request is billed according to the bucket where the target object is stored.
  - The used storage capacity is billed according to the bucket where the target object is stored.
- Predetermined request headers
  - You can include any number of the four predetermined request headers ( x-oss-copy-source-if-match , x-oss-copy-source-if-none-match , x-oss-copy-source-if-unmodified-since ,

- and `x-oss-copy-source-if-modified-since` ) in a request at the same time. For more information, see [Detail analysis](#) in `GetObject`.
- The request headers used in copy operations start with `x-oss-`. Therefore, these headers must be added into the signature string.

## Examples

### • Example 1

#### Request example:

```
PUT / copy_oss . jpg HTTP / 1 . 1
Host : oss - example . oss - cn - hangzhou . aliyuncs . com
Date : Fri , 24 Feb 2012 07 : 18 : 48 GMT
x - oss - storage - class : Archive
x - oss - copy - source : / oss - example / oss . jpg
Authorization : OSS qn6qrrqxo2 oawuk53otf jbyc : gmnwPKuu20
LQEjd + iPkL259A + n0 =
```

#### Response example:

```
HTTP / 1 . 1 200 OK
x - oss - request - id : 559CC9BDC7 55F95A6448 5981
Content - Type : application / xml
Content - Length : 193
Connection : keep - alive
Date : Fri , 24 Feb 2012 07 : 18 : 48 GMT
Server : AliyunOSS
<? xml version = " 1 . 0 " encoding = " UTF - 8 " ? >
< CopyObject Result xmlns = " http : / / doc . oss - cn - hangzhou .
aliyuncs . com ">
  < LastModified > Fri , 24 Feb 2012 07 : 18 : 48 GMT < /
  LastModified >
  < ETag > " 5B3C1A2E05 3D763E1B00 2CC607C5A0 FE " < / ETag >
< / CopyObject Result >
```

### • Example 2

#### Request example:

```
PUT / test % 2FAK . txt HTTP / 1 . 1
Host : tesx . oss - cn - zhangjiako u . aliyuncs . com
Accept - Encoding : identity
User - Agent : aliyun - sdk - python / 2 . 6 . 0 ( Windows / 7 /
AMD64 ; 3 . 7 . 0 )
Accept : */*
Connection : keep - alive
x - oss - copy - source : / test / AK . txt
date : Fri , 28 Dec 2018 09 : 41 : 55 GMT
authorization : OSS qn6qrrqxo2 oawuk53otf jbyc : gmnwPKuu20
LQEjd + iPkL259A + n0 =
Content - Length : 0
```

#### Response example:

```
HTTP / 1 . 1 200 OK
```

```

Server : AliyunOSS
Date : Fri , 28 Dec 2018 09 : 41 : 56 GMT
Content - Type : application / xml
Content - Length : 184
Connection : keep - alive
x - oss - request - id : 5C25EFE446 2CE00EC6D8 7156
ETag : " F2064A169E E92E9775EE 5324D0B168 2E "
x - oss - hash - crc64ecma : 1275300285 9196105360
x - oss - server - time : 150

<? xml version =" 1 . 0 " encoding =" UTF - 8 "? >
< CopyObject Result >
  < ETag >" F2064A169E E92E9775EE 5324D0B168 2E "</ ETag >
  < LastModifi ed > 2018 - 12 - 28T09 : 41 : 56 . 000Z </
LastModifi ed >
</ CopyObject Result >

```

**Note:**

x-oss-hash-crc64ecma indicates the 64-bit CRC value of the object. This value is calculated based on the [ECMA-182](#) standard. An object generated in a COPY operation may not have a CRC64 value.

## 4.3 AppendObject

AppendObject is used to upload a file by appending the file to an existing object.

An object created with the AppendObject operation is an appendable object, and an object uploaded with the PutObject operation is a normal object.

**Note:**

- You cannot use AppendObject to upload a file to an object protected by the WORM policy.
- You cannot use KMS to encrypt appendable objects on the server by specifying CMK IDs for them.

### Association with other operations

| Operations                 | Relationship   |
|----------------------------|--|
| <a href="#">PutObject</a>  | If you perform a PutObject operation on an existing appendable object, the appendable object is overwritten by a new normal object.  |
| <a href="#">HeadObject</a> | If you perform a HeadObject operation on an existing appendable object, then x-oss-next-append-position, x-oss-hash-crc64ecma, and x-oss-object-type are returned. The x-oss-object-type of the appendable object is Appendable. |



| Operations                 | Relationship  |
|----------------------------|---|
| <a href="#">GetBucket</a>  | In the response to a GetBucket request, the x-oss-object-type of the appendable object is set to Appendable.  |
| <a href="#">CopyObject</a> | You can neither use CopyObject to copy an appendable object , nor change the server-side encryption method of this object . However, you can use CopyObject to modify the custom metadata of an object. |

### Request syntax

```
POST / ObjectName ? append & position = Position HTTP / 1 . 1
Content - Length : ContentLength
Content - Type : ContentType
Host : BucketName . oss . aliyuncs . com
Date : GMT Date
Authorization : SignatureValue
```

### Parameters in an AppendObject request

An AppendObject request must include the append and position parameters, which are both CanonicalizedResource and must be included in the signature.

- `append`

This parameter indicates that the request is sent to perform an AppendObject operation.

- `position`

This parameter specifies the position from where the append operation starts. The value of position in the first AppendObject operation must be 0, and the value of position in the subsequent operation is the current object length. For example , if the value of position specified in the first AppendObject request is 0, and the value of content-length is 65536, the value of position specified in the second AppendObject request must be set to 65536.

Each time after an AppendObject operation succeeds, x-oss-next-append-position in the response header specifies the position of the next AppendObject request.

Note the following when setting position:



- If the value of position is 0 and an object with the same name does not exist, you can set headers (such as x-oss-server-side-encryption) in the AppendObject request in the same way as you do in a PutObject request. If you add a correct x-oss-server-side-encryption header in an AppendObject request in which the

value of position is 0, the x-oss-server-side-encryption header is also included in the response header. You can initiate a CopyObject request to modify the metadata of the object in subsequent operations.

- If the value of position is 0 and an appendable object with the same name does not exist, or if the length of an appendable object with the same name is 0, the AppendObject operation is successful. Otherwise, the system determines that the position and object length do not match and returns a PositionNotEqualToLength error code.
- The length limit of an object generated by an AppendObject operation is the same as that of an object generated by a PutObject operation. Each time after an AppendObject operation is performed, the last modification time of this object is updated.
- If the position value is correct and content with a length of 0 is appended to an existing appendable object, the status of the object does not change.

#### Request headers

| Header                | Type   | Description  |
|-----------------------|--------|--|
| Cache - Control       | String | Specifies the Web page caching behavior for the object. For more information, see <a href="#">RFC2616</a> .<br>Default value: none   |
| Content - Disposition | String | Specifies the name of the object when the object is downloaded. For more information, see <a href="#">RFC2616</a> .<br>Default value: none   |
| Content - Encoding    | String | Specifies the content encoding format of the object. For more information, see <a href="#">RFC2616</a> .<br>Default value: none  |
| Content - MD5         | String | Content-MD5 is a string calculated by the MD5 algorithm. This header is used to check whether the message content is consistent with the sent content. The value of Content-MD5 can be obtained as follows : Calculate a 128-bit number based on the message content, rather than the header, and then base64-encode the number.<br>Default value: none<br>Restriction: none |

| Header                               | Type    | Description  |
|--------------------------------------|---------|--|
| Expires                              | Integer | Specifies the expiration time. For more information, see <a href="#">RFC2616</a> .<br>Default value: none  |
| x - oss - server - side - encryption | String  | Specifies the server-side encryption algorithm.<br>Valid values: AES256 or KMS<br><br><div>  <b>Note:</b><br/>           You must enable KMS (Key Management Service) in the console before you can use the KMS encryption algorithm. Otherwise, a KmsServiceNotEnabled error is returned.         </div>   |
| x - oss - object - acl               | String  | Specifies the ACL for the object.<br>Valid values: public - read , private , and public - read - write   |
| x - oss - storage - class            | String  | Specifies the storage class of the object.<br>Values: <ul style="list-style-type: none"> <li>Standard</li> <li>IA</li> <li>Archive</li> </ul> Supported interfaces: PutObject, InitMultiPartUpload, AppendObject, PutObjectSymlink, and CopyObject<br><br><div>  <b>Note:</b> <ul style="list-style-type: none"> <li>The status code 400 Bad Request is returned if the value of StorageClass is invalid. Error description: InvalidArgument.</li> <li>If you specify the value of x-oss-storage-class when uploading an object to a bucket, the storage class of the uploaded object is the specified value of x-oss-storage-class regardless of the storage class of the bucket. For example, if you specify the value of x-oss-storage-class to Standard when uploading an object to a bucket of the IA storage class, the storage class of the object is Standard.</li> <li>This header takes effect only if you specify it when you perform the AppendObject operation for the first time.</li> </ul> </div> |

## Response headers

| Header                     | Type           | Description  |
|----------------------------|----------------|--|
| x-oss-next-append-position | 64-bit integer | Specifies the position that must be provided in the next request, that is, the current object length. This header is returned when a successful message is returned for an AppendObject request, or when a 409 error occurs because the position and the object length do not match. |
| x-oss-hash-crc64ecma       | 64-bit integer | Specifies the 64-bit CRC value of the object. This value is calculated according to the <a href="#">ECMA-182</a> .   |

## CRC64 calculation method

The CRC value of an appendable object is calculated according to [ECMA-182](#). You can calculate the CRC64 in the following methods:

- Calculate using boost CRC module:

```
typedef boost::crc_optimal l < 64, 0x42F0E1EB A9EA3693UL L
, 0xffffffff ffffffffUL L, 0xffffffff ffffffffUL L, true
, true > boost_ecma;

uint64_t do_boost_crc ( const char * buffer, int length
)
{
    boost_ecma crc;
    crc.process_bytes ( buffer, length );
    return crc.checksum ();
}
```

- Calculate using the Python crcmod:

```
do_crc64 = crcmod.mkCrcFun ( 0x142F0E1E BA9EA3693L, initCrc
= 0L, xorOut = 0xffffffff ffffffffL, rev = True )

print do_crc64 ( " 123456789 " )
```

## Example

## Request example:

```
POST / oss . jpg ? append & position = 0 HTTP / 1 . 1
Host : oss - example . oss . aliyuncs . com
Cache - control : no - cache
Expires : Wed , 08 Jul 2015 16 : 57 : 01 GMT
Content - Encoding : utf - 8
x - oss - storage - class : Archive
Content - Disposition : attachment ; filename = oss_download .
jpg
Date : Wed , 08 Jul 2015 06 : 57 : 01 GMT
```

```

Content - Type : image / jpg
Content - Length : 1717
Authorization : OSS qn6qrrqxo2 oawuk53otf jbyc : kZoYNv66bs
mc10 + dcGKw5x2PR rk =
[ 1717 bytes of object data ]

```


### Response example:

```

HTTP / 1.1 200 OK
Date : Wed, 08 Jul 2015 06:57:01 GMT
ETag : "0F7230CAA4 BE94CCBDC9 9C55000000 00"
Connection : keep-alive
Content - Length : 0
Server : AliyunOSS
x-oss-hash-crc64ecma : 1474161709 5266562575
x-oss-next-append-position : 1717
x-oss-request-id : 559CC9BDC7 55F95A6448 5981

```

### Error messages

| Error message            | HTTP status code | Description   |
|--------------------------|------------------|---|
| ObjectNotAppendable      | 409              | You cannot perform AppendObject operations on a non-appendable object.  |
| PositionNotEqualToLength | 409              | <p>The value of position does not match the current object length. You can obtain the position for the next operation from the response header x-oss-next-append-position and initiate a request again.</p> <div>  <b>Note:</b> <ul style="list-style-type: none"> <li>Although multiple requests may be sent concurrently, even if you set the value of x-oss-next-append-position in one request, the request may still fail because the value is not updated immediately.</li> <li>The PositionNotEqualToLength error message is returned if the value of position is 0 and the length of an appendable object with the same name is not 0.</li> </ul> </div> |

## 4.4 HeadObject

HeadObject is used to return the meta information of a certain object without returning the file content.

### Request syntax

```
HEAD / ObjectName HTTP / 1 . 1
Host : BucketName / oss - cn - hangzhou . aliyuncs . com
Date : GMT Date
Authorizat ion : SignatureV alue
```

### Request header

| Name                    | Type   | Description  |
|-------------------------|--------|--|
| If - Modified - Since   | String | If the specified time is earlier than the actual modification time, the system returns the 200 OK message and the object metadata; otherwise, the system returns the 304 Not Modified message.<br>Default: None                      |
| If - Unmodified - Since | String | If the specified time is same as or later than the actual file modification time, the system returns the 200 OK message and the object metadata; otherwise, the system returns the 412 Precondition Failed message.<br>Default: None |

| Name              | Type   | Description  |
|-------------------|--------|--|
| If - Match        | String | If the expected ETag that is introduced matches the ETag of the object, the system returns the 200 OK message and the object metadata; otherwise, the system returns the 412 Precondition Failed message.<br>Default: None |
| If - None - Match | String | If the introduced ETag does not match the ETag of the object, the system returns the 200 OK message and the object metadata; otherwise, the system returns the 304 Not Modified message.<br>Default: None                  |

#### Detail analysis

- After the Head Object request is sent, no message body is returned even if the system returns the 200 OK message or an error message.
- The If-Modified-Since, If-Unmodified-Since, If-Match, and If-None-Match query conditions can be set in the header of the Head Object request. For the detailed setting rules, see the related fields in the Get Object request. If no modification is made, the system returns the 304 Not Modified message.
- If you upload the user meta prefixed with x-oss-meta- when sending a Put Object request, for example, x-oss-meta-location, the user meta is returned.
- If the file does not exist, the system returns Error 404 Not Found.
- If this object is entropy encrypted on the server, the system returns x-oss-server-side-encryption in the header of the response to the Head Object request. The value of x-oss-server-side-encryption indicates the server-side encryption algorithm of the object.
- If the file type is symbolic link, in the response header, `Content - Length` , `ETag` , and `Content - Md5` are metadata of the target file, `Last - Modified`

is the maximum value of the target file and symbolic link, and others are metadata of symbolic links.

- If the file type is symbolic link and the target file does not exist, the system returns Error 404 Not Found. The error code is “SymlinkTargetNotExist” .
- If the file type is symbolic link and the target file type is symbolic link, the system returns Error 400 Bad request. The error code is “InvalidTargetType” .
- If the bucket type is Archive and the Restore request has been submitted, the Restore state of Object is indicated by x-oss-restore in the response header.
  - If the Restore request is not submitted or times out, the field is not returned.
  - If the Restore request has been submitted and does not time out, the value of x-oss-restore returned is ongoing-request=” true” .
  - If the Restore request has been submitted and completed, the value of x-oss-restore returned is ongoing-request=” false” , expiry-date=” Sun, 16 Apr 2017 08:12:33 GMT” . Where the expiry-date refers to the expiry date of the readable state of the restored file.

## Example

### Request example:

```
HEAD / oss . jpg HTTP / 1 . 1
Host : oss - example . oss - cn - hangzhou . aliyuncs . com
Date : Fri , 24 Feb 2012 07 : 32 : 52 GMT
Authorization : OSS qn6qrrqxo2 oawuk53otf jbyc : JbzF2LxZUt
anlJ5dLA09 2wpDC / E =
```

### Return example:

```
HTTP / 1 . 1 200 OK
x - oss - request - id : 559CC9BDC7 55F95A6448 5981
x - oss - object - type : Normal
x - oss - storage - class : Archive
Date : Fri , 24 Feb 2012 07 : 32 : 52 GMT
Last - Modified : Fri , 24 Feb 2012 06 : 07 : 48 GMT
ETag : " fba9dede5f 27731c9771 645a398633 28 "
Content - Length : 344606
Content - Type : image / jpg
Connection : keep - alive
Server : AliyunOSS
```

Example of a request when the Restore request has been submitted but not completed :

```
HEAD / oss . jpg HTTP / 1 . 1
Host : oss - archive - example . oss - cn - hangzhou . aliyuncs . com
Date : Sat , 15 Apr 2017 07 : 32 : 52 GMT
```



```
Authorization : OSS e1Unnbm1rg dnpI : KKxkdNrUBu 2t1kqlDh0M
LbDb99I =
```

### Return example:

```
HTTP / 1 . 1 200 OK
x - oss - request - id : 58F71A1645 29F18D7F00 0045
x - oss - object - type : Normal
x - oss - storage - class : Archive
x - oss - restore : ongoing - request =" true "
Date : Sat , 15 Apr 2017 07 : 32 : 52 GMT
Last - Modified : Sat , 15 Apr 2017 06 : 07 : 48 GMT
ETag : " fba9dede5f 27731c9771 645a398633 28 "
Content - Length : 344606
Content - Type : image / jpg
Connection : keep - alive
Server : AliyunOSS
```

### Example of a request when the Restore request has been submitted and completed:

```
HEAD / oss . jpg HTTP / 1 . 1
Host : oss - archive - example . oss - cn - hangzhou . aliyuncs .
com
Date : Sat , 15 Apr 2017 09 : 35 : 51 GMT
Authorization : OSS e1Unnbm1rg dnpI : 2lqtGJ + ykDVmdu606
FMJnn + WuBw =
```

### Return example:

```
HTTP / 1 . 1 200 OK
x - oss - request - id : 58F7253445 29F18D7F00 0055
x - oss - object - type : Normal
x - oss - storage - class : Archive
x - oss - restore : ongoing - request =" false ", expiry - date ="
Sun , 16 Apr 2017 08 : 12 : 33 GMT "
Date : Sat , 15 Apr 2017 09 : 35 : 51 GMT
Last - Modified : Sat , 15 Apr 2017 06 : 07 : 48 GMT
ETag : " fba9dede5f 27731c9771 645a398633 28 "
Content - Length : 344606
```

## 4.5 PostObject

The PostObject operation is used to upload an object to a specified bucket using the HTML form.

### Post object

#### · Request syntax

```
POST / HTTP / 1 . 1
Host : BucketName . oss - cn - hangzhou . aliyuncs . com
User - Agent : browser_data
Content - Length : ContentLength
Content - Type : multipart / form - data ; boundary = 9431149156
168
-- 9431149156 168
```

```

Content - Dispositio n : form - data ; name =" key "
key
-- 9431149156 168
Content - Dispositio n : form - data ; name =" success_ac
tion_redir ect "
success_re direct
-- 9431149156 168
Content - Dispositio n : form - data ; name =" Content -
Dispositio n "
attachment ; filename = oss_downlo ad . jpg
-- 9431149156 168
Content - Dispositio n : form - data ; name =" x - oss - meta -
uuid "
myuuid
-- 9431149156 168
Content - Dispositio n : form - data ; name =" x - oss - meta -
tag "
mytag
-- 9431149156 168
Content - Dispositio n : form - data ; name =" OSSAccessK eyId
"
access - key - id
-- 9431149156 168
Content - Dispositio n : form - data ; name =" policy "
encoded_po licy
-- 9431149156 168
Content - Dispositio n : form - data ; name =" Signature "
signature
-- 9431149156 168
Content - Dispositio n : form - data ; name =" file "; filename
=" MyFilename . jpg "
Content - Type : image / jpeg
file_conte nt
-- 9431149156 168
Content - Dispositio n : form - data ; name =" submit "
Upload to OSS
-- 9431149156 168 --

```


• Request header

| Header          | Type   | Required?         | Description   |
|-----------------|--------|-------------------|---|
| OSSAccessK eyId | String | Yes in some cases | Specifies the AccessKey ID of the bucket owner.<br>Default value: none<br>Restriction: This field is required when the bucket does not allow public-read-write and when the OSSAccessKeyId (or Signature) form field is provided. |

| Header | Type   | Required?         | Description   |
|--------|--------|-------------------|---|
| policy | String | Yes in some cases | <p>Specifies the validity of the fields in the request. A request that does not contain the policy field is treated as an anonymous request, and can only access buckets that allow public-read-write.</p> <p>Default value: none</p> <p>Restriction: This form field is required when the bucket does not allow public-read-write, or when the OSSAccessKeyId (or Signature) form field is provided.</p> |

| Header   | Type   | Required?         | Description   |
|--|--------|-------------------|---|
| Signature  | String | Yes in some cases | Specifies the signature information that is computed based on the Access Key Secret and Policy. OSS checks the signature information to verify validity of the Post Object request. For more information, see 5.7.4.2 Post Signature.<br>Default value: none<br>Restriction: This form field is required when the bucket does not allow public-read-write, or when the OSSAccessKeyId (or Policy) form field is provided. |
| Cache - Control , Content - Type , Content - Dispositio n , Content - Encoding , Expires | String | No                | HTTP request headers. For more information, see <a href="#">PutObject</a> .<br>Default value: none  |

| Header                  | Type   | Required? | Description  |
|-------------------------|--------|-----------|--|
| file                    | String | Yes       | Specifies the file or text content. It must be the last field in the form. The browser automatically sets the Content-Type based on the file type and overwrites the user setting. Only one file can be uploaded to OSS at a time.<br>Default value: none                                    |
| key                     | String | Yes       | Specifies the name of the uploaded object. If the object name includes a path, such as a/b/c/b.jpg, OSS automatically creates the corresponding directory.<br>Default value: none  |
| success_action_redirect | String | No        | Specifies the URL to which the client is redirected after successful upload. If this form field is not specified, the returned result is specified by success_action_status. If upload fails, OSS returns an error code, and the client is not redirected to any URL.<br>Default value: none |

| Header                | Type   | Required? | Description   |
|-----------------------|--------|-----------|---|
| success_action_status | String |           | <p>Specifies the status code returned to the client after the previous successful upload if success_action_redirect is not specified. Default value: none Valid values: 200, 201, and 204 (default)</p> <div>  <b>Note:</b> <ul style="list-style-type: none"> <li>• If the value of this field is set to 200 or 204, OSS returns an empty file and the 200 or 204 status code.</li> <li>• If the value of this field is set to 201, OSS returns an XML file and the 201 status code.</li> <li>• If this field is not specified or set to an invalid value, OSS returns an empty file and the 204 status code.</li> </ul> </div> |
| x-oss-meta-*          | String | No        | <p>Specifies the user meta value set by the user. OSS does not check or use this value. Default value: none</p>   |

| Header  | Type   | Required? | Description   |
|---|--------|-----------|---|
| x - oss - server - side - encryption            | String | No        | Specifies the server-side encryption algorithm when OSS creates an object.<br>Valid value:<br>AES256                                    |
| x - oss - server - side - encryption - key - id | String | No        | Specifies the primary key managed by KMS. This parameter is valid when the value of x - oss - server - side - encryption is set to KMS. |
| x - oss - object - acl                          | String | No        | Specifies the ACL for the created object.<br>Valid values:<br>public - read , private , and public - read - write                       |

| Header                     | Type   | Required? | Description   |
|----------------------------|--------|-----------|---|
| x - oss - security - token | String | No        | If STS temporary authorization is used for this access, you must specify the item to be the SecurityToken value. At the same time, OSSAccessKeyId must use a paired temporary AccessKeyId. The signature calculation is consistent with the general AccessKeyId signature.<br>Default value: none |

### Response header

| Header                               | Type   | Description  |
|--------------------------------------|--------|--|
| x - oss - server - side - encryption | String | If x-oss-server-side-encryption is specified in the request, the response contains this header , which indicates the encryption algorithm used . |

### Response elements

| Parameter    | Type      | Description   |
|--------------|-----------|---|
| PostResponse | Container | Specifies the container that saves the result of the PostObject request.<br>Sub-elements: Bucket, ETag, Key, and Location |
| Bucket       | String    | Specifies the bucket name.<br>Parent element: PostResponse  |



| Parameter | Type   | Description  |
|-----------|--------|--|
| ETag      | String | Specifies the entity tag (ETag) that is created when an object is generated. For an object created by Post Object, the ETag value is the UUID of the object, and can be used to check whether the content of the object has changed.<br>Parent element: PostResponse |
| Location  | String | Specifies the URL of the newly created object.<br>Parent element: PostResponse   |

### Detail analysis

- To perform the Post Object operation, you must have the permission to write the bucket. If the bucket allows public-read-write, you can choose not to upload the signature information. Otherwise, signature verification must be performed on the Post Object operation. Unlike Put Object, Post Object uses AccessKeySecret to compute the signature for the policy. The computed signature string is used as the value of the Signature form field. OSS checks this value to verify validity of the signature.
- No matter whether the bucket allows public-read-write, once any one of the OSSAccessKeyId, Policy, and Signature form fields is uploaded, the remaining two form fields are required. If the remaining two form fields are missing, OSS returns the error code: InvalidArgument.
- Form encoding submitted by the Post Object operation must be "multipart/form-data". That is, Content-Type in the header must be in the `multipart / form - data ; boundary = xxxxxx` format, where boundary is the boundary string.
- The URL of the submitted form can be the domain name of the bucket. It is not necessary to specify the object in the URL. The request uses `POST / HTTP / 1 . 1` but not `POST / ObjectName HTTP / 1 . 1`.
- The form and policy must be encoded with UTF-8.
- If you have uploaded the Content-MD5 request header, OSS calculates the body's Content-MD5 and check if the two are consistent. If the two are different, the error code InvalidDigest is returned.
- If the Post Object request contains the Header signature or URL signature, OSS does not check these signatures.

- If the Put Object request carries a form field prefixed with x-oss-meta-, the form field is treated as the user meta, for example, x-oss-meta-location. A single object can have multiple similar parameters, but the total size of all user meta cannot exceed 8 KB.
- The total length of the body in the Post Object request cannot exceed 5 GB. When the file length is too large, the system returns the error code: EntityTooLarge.
- If the x-oss-server-side-encryption header is specified when you upload an object, the value of this header must be set to AES256 or KMS. Otherwise, a 400 error is returned with the error code: InvalidEncryptionAlgorithmError. After this header is specified, the response header also contains this header, and OSS stores the encryption algorithm of the uploaded object. When this object is downloaded, the response header contains x-oss-server-side-encryption, the value of which is set to the encryption algorithm of this object.
- Form fields are not case-sensitive, but their values are case-sensitive.

## Examples

- Request example:

```
POST / HTTP / 1 . 1
Host : oss - example . oss - cn - hangzhou . aliyuncs . com
Content - Length : 344606
Content - Type : multipart / form - data ; boundary = 9431149156
168
-- 9431149156 168
Content - Dispositio n : form - data ; name =" key "
/ user / a / objectName . txt
-- 9431149156 168
Content - Dispositio n : form - data ; name =" success_ac
tion_statu s "
200
-- 9431149156 168
Content - Dispositio n : form - data ; name =" Content -
Dispositio n "
content_di sposition
-- 9431149156 168
Content - Dispositio n : form - data ; name =" x - oss - meta -
uuid "
uuid
-- 9431149156 168
Content - Dispositio n : form - data ; name =" x - oss - meta -
tag "
metadata
-- 9431149156 168
Content - Dispositio n : form - data ; name =" OSSAccessK eyId
"
44CF959000 6BF252F707
-- 9431149156 168
Content - Dispositio n : form - data ; name =" policy "
eyJleHBpcm F0aW9uIjoI MjAxMy0xMi 0wMVQxMjow MDowMFoiLC
Jjb25kaXRp b25zIjpbWy Jjb250ZW50 LWxlbmd0aC 1yYW5nZSIs
```

```

IDAsIDEwND g1NzYwXSx7 ImJ1Y2tldC I6ImFoYWhh In0sIHsiQS
I6ICJhIn0s eyJrZXkiOi AiQUJDIn1d fQ ==
-- 9431149156 168
Content - Dispositio n : form - data ; name =" Signature "
kZoYNv66bs mc10 + dcGKw5x2PR rk =
-- 9431149156 168
Content - Dispositio n : form - data ; name =" file "; filename
=" MyFilename . txt "
Content - Type : text / plain
abcdefg
-- 9431149156 168
Content - Dispositio n : form - data ; name =" submit "
Upload to OSS
-- 9431149156 168 --

```

- **Response example:**

```

HTTP / 1 . 1 200 OK
x - oss - request - id : 61d2042d - 1b68 - 6708 - 5906 -
33d8192136 2e
Date : Fri , 24 Feb 2014 06 : 03 : 28 GMT
ETag : 5B3C1A2E05 3D763E1B00 2CC607C5A0 FE
Connection : keep - alive
Content - Length : 0
Server : AliyunOSS

```

## Post Policy

The policy form field requested by POST is used to verify the validity of the request. The policy is a JSON text encoded with UTF-8 and Base64. It states the conditions that a Post Object request must meet. The post form field is optional for uploading public-read-write buckets. However, we strongly recommend you use this field to limit POST requests.

### Policy example

```

{ " expiration ": " 2014 - 12 - 01T12 : 00 : 00 . 000Z ",
  " conditions ": [
    { " bucket ": " johnsmith " },
    [ " starts - with ", "$ key ", " user / eric /" ]
  ]
}

```

In a PostObject request, the policy must contain expiration and conditions.

### Expiration

Expiration specifies the expiration time of the policy, which is expressed in ISO8601 GMT. For example, "2014-12-01T12:00:00.000Z" means that the Post Object request must be sent before 12:00 on December 1, 2014.

### Conditions

Conditions is a list that specifies the valid values of form fields in the Post Object request. Note: The value of a form field is extended after OSS checks the policy. Therefore, the valid value of the form field set in the policy is equivalent to the value of the form field before extension. The following table lists the conditions supported by the policy:

| Parameter  | Description   |
|--|---|
| content - length - range   | Specifies the acceptable maximum and minimum sizes of the uploaded file. This condition supports the content-length-range match mode.   |
| Cache - Control , Content - Type , Content - Dispositio n , Content - Encoding , Expires | HTTP request headers. This condition supports the exact match and starts-with match modes.  |
| key  | Specifies the object name of the uploaded file. This condition supports the exact match and starts-with match modes.  |
| success_ac tion_redir ect  | Specifies the URL to which the client is redirected after successful upload. This condition supports the exact match and starts-with match modes.                             |
| success_ac tion_statu s  | Specifies the status code returned after successful upload if success_action_redir ect is not specified. This condition supports the exact match and starts-with match modes. |
| x - oss - meta -*  | Specifies the meta value set by the user. This condition supports the exact match and starts-with match modes.  |

If the PostObject request contains extra form fields, OSS adds these fields to the conditions of the policy and checks their validity.

#### Condition match modes

| Condition match modes | Description  |
|-----------------------|--|
| Exact match           | The value of a form field must be exactly the same as the value declared in the conditions. For example, if the value of the key form field must be a, the conditions must be: { "key" : "a" }, or: [ "eq" , "\$key" , "a" ] |
| Starts With           | The value of a form field must start with the specified value. For example, if the value of key must start with user/user1, the conditions must be: [ "starts-with" , "\$key" , "user/user1" ]                               |
| Specified file size   | Specifies the range of the allowed file size . For example, if the acceptable file size is 1 to 10 bytes, the conditions must be: ["content-length-range", 1, 10]  |

### Escape characters

In the policy form field of the Post Object request, \$ is used to indicate a variable . Therefore, to describe \$, the escape character must be used. In addition, some characters in JSON strings are escaped. The following table describes characters in the JSON string of the policy form field of a Post Object request.

| Escape characters | Description            |
|-------------------|------------------------|
| \/                | Slash                  |
| \                 | Backslash              |
| \"                | Double quotation marks |
| \\$               | Dollar sign            |
| Space             | Space                  |
| \f                | Form feed              |
| \n                | Newline                |
| \r                | Carriage return        |
| \t                | Horizontal tab         |
| \uxxxx            | Unicode character      |

## Post Signature

For a verified Post Object request, the HTML form must contain policy and signature . Policy specifies which values are acceptable in the request. The procedure for computing signature is as follows:

1. Create a UTF-8 encoded policy.
2. Encode the policy with Base64. The encoding result is the value of the policy form field, and this value is used as the string to be signed.
3. Use AccessKeySecret to sign the string. The signing method is the same as the computing method of the signature in the Header, that is, replacing the string to be signed with the policy form field.

## 4.6 Callback

To enable OSS to return callback information about an object to an application server after the object is uploaded to OSS, you just need to add a callback parameter in the upload request sent to OSS. This topic describes the implementation of upload callback in details.

### Background

- APIs that support upload callback include: [PutObject](#), [PostObject](#), and [CompleteMultipartUpload](#).
- Regions that support upload callback include: China North 2 (Beijing), China East 1 (Hangzhou), China North 1 (Qingdao), China East 2 (Shanghai), China East 2 (Finance Cloud), China South 1 (Finance Cloud), China South 1 (Shenzhen), Hong Kong, China North 5 (Hohhot), China North 3 (Zhangjiakou), Middle East 1 (Dubai ), Asia Pacific NE 1 (Tokyo), EU Central 1 (Frankfurt), Asia Pacific SE 1 (Singapore), US East 1 (Virginia), US West 1 (Silicon Valley), Asia Pacific SE 2 (Sydney), and Asia Pacific SE 3 (Kuala Lumpur).
- For more information about upload callback, see [Principle](#).

### Step 1: Construct parameters.

- Construct a callback parameter.

A callback parameter is a base64-encoded string (field) in JSON format. To construct a callback parameter, it is important to specify the URL of the server to

which the callback information is returned (callbackUrl) and the content of the callback information (callbackBody).

The following table describes the JSON fields included in a callback parameter.

| Field       | Description  | Required |
|-------------|--|----------|
| callbackUrl | <ul style="list-style-type: none"> <li>- After an object is uploaded, OSS sends a callback request using the POST method to this URL. The body of the request is the content specified in callbackBody. This URL returns an <code>HTTP / 1 . 1 200 OK</code> response only when the following conditions are met: 1. The body of the callback request is in JSON format. 2. The Content-Length header of the request must be a valid value smaller than 3 MB.</li> <li>- You can set five URLs in a request in maximum by separating them by semicolons (;). OSS sends requests to each URL until the first success response is returned.</li> <li>- If no URL is configured or the value of this field is null, OSS determines that the callback function is not configured.</li> <li>- HTTPS IP addresses are supported.</li> <li>- To ensure that Chinese characters can be correctly processed, the callbackUrl must be encoded. For example, if the value of callbackUrl is <code>http :// example . com / Chinese characters . php ? key = value &amp; Chinese Name = Chinese Value</code>, it must be encoded into a JSON string, such as <code>http :// example . com /% E4 % B8 % AD % E6 % 96 % 87 . php ? key = value &amp;% E4 % B8 % AD % E6 % 96 % 87 % E5 % 90 % 8D % E7 % A7 % B0 =% E4 % B8 % AD % E6 % 96 % 87 % E5 % 80 % BC</code>.</li> </ul> | Yes      |

| Field            | Description  | Required |
|------------------|--|----------|
| callbackHost     | <ul style="list-style-type: none"> <li>- Indicates the value of the Host header in the callback request. This field is valid only when the callbackUrl is specified.</li> <li>- If this field is not specified, the hosts in the URLs specified in the callbackUrl field are resolved and specified as the value of this field.</li> </ul>   | No       |
| callbackBody     | <ul style="list-style-type: none"> <li>- Indicates the value of the callback request body, for example, as key=\$(key)&amp;etag=\$(etag)&amp;my_var=\$(x:my_var).</li> <li>- System variables, custom variables, and constants are supported for this field. Custom variables are passed through the callback-var parameter in PutObject and CompleteMultipart operations and through form fields in PostObject operations.</li> </ul>   | Yes      |
| callbackBodyType | <ul style="list-style-type: none"> <li>- Indicates the Content-Type header in the callback request. This field supports two values: application/x-www-form-urlencoded and application/json, in which application/x-www-form-urlencoded is the default value.</li> <li>- If the value of callbackBodyType is application/x-www-form-urlencoded, variables in callbackBody are replaced by the encoded URLs. If the value of callbackBodyType is application/json, the variables are replaced in JSON format.</li> </ul> | No       |

Examples of the JSON fields included in a callback parameter are as follows:

```
{
  "callbackUrl": "121.101.166.30/test.php",
  "callbackHost": "oss-cn-hangzhou.aliyuncs.com",
  "callbackBody": "{\" mimeType \": \"${ mimeType },\" size \": \"${ size }\"",
  "callbackBodyType": "application/json"
}
```



```
}
```

```
{
  " callbackUr  l ":" 121 . 43 . 113 . 8 : 23456 / index . html ",
  " callbackBo  dy ":" bucket =${ bucket }& object =${ object }& etag
    =${ etag }& size =${ size }& mimeType =${ mimeType }& imageInfo .
    height =${ imageInfo . height }& imageInfo . width =${ imageInfo .
    width }& imageInfo . format =${ imageInfo . format }& my_var =${ x
    : my_var }"
}
```

The following table describes configurable system parameters in callbackBody.

| System parameters | Description   |
|-------------------|---|
| bucket            | Indicates the bucket where the request object is stored.  |
| object            | Indicates the requested object.   |
| etag              | Indicates the ETag of the object, that is, the ETag field returned to the user who sends the request.             |
| size              | Indicates the size of the requested object, which is the total object size in CompleteMultipartUpload operations. |
| mimeType          | Indicates the resource type. For example, the resource type of JPEG images is image/jpeg.                         |
| imageInfo.height  | Indicates the height of an image.   |
| imageInfo.width   | Indicates the width of an image.  |
| imageInfo.format  | Indicates the format of an image, such as jpg or png.   |



**Note:**

Only an image object supports the imageinfo parameter. The values of imageInfo.height, imageInfo.width, imageInfo.format are null if the object is not an image.

- Construct custom parameters using callback-var.

You can configure custom parameters by using the callback-var parameter. A custom parameter is a key-value map. You can add required parameters to the map. When a POST callback request is initiated, OSS adds these custom parameters and the system parameters described in the preceding section to the body of the POST

request, so that these parameters can be easily obtained by the user who sends the callback request.

You can construct a custom parameter in the same way as you construct a callback parameter. A custom parameter is also a string in JSON format, which is a key-value map including all custom parameters.



**Note:**

The key of a custom parameter must start with "x:" and be lower-cased. Otherwise, OSS returns an error.

Assume that you need to configure two custom parameters `x:var1` and `x:var2`, and the values of the two parameters are `value1` and `value2` respectively. The constructed JSON strings are as follows:

```
{
  " x : var1 ":" value1 ",
  " x : var2 ":" value2 "
}
```



**Note:**

If the input callback parameter or callback-var parameter is invalid, a 400 error is returned with the `InvalidArgument` error code. This occurs in the following scenarios:

- URLs and headers are input at the same time to the callback parameter (`x-oss-callback`) or the callback-var parameter (`x-oss-callback-var`) in `PutObject` and `CompleteMultipartUpload` operations.
- The size of the callback or callback-var parameter (this does not occur in `PostObject` operations because the callback-var parameter is not available in `PostObject` operations) exceeds 5 KB.
- The callback or callback-var parameter is not base64-encoded or is not in the valid JSON format after being decoded.
- The `callbackUrl` field decoded from the callback parameter includes more than five URLs, or the port in the URL is invalid, for example:

```
{" callbackUrl ":" 10 . 101 . 166 . 30 : test ",
  " callbackBody ":" test "}
```

- The `callbackBody` field decoded from the callback parameter is null.

- The value of `callbackBodyType` decoded from the `callback` parameter is not `applicatio n / x - www - form - urlencoded` or `applicatio n / json`.
- The variables in the `callbackBody` field decoded from the `callback` parameter are not in the valid format, that is, `${var}`.
- The `callback-var` parameter is not in the expected JSON format, that is, `{" x : var1 ":" value1 "," x : var2 ":" value2 "...}`.

#### Step 2: Construct a callback request.

After constructing the `callback` and `callback-var` parameters, you must add the parameters to the callback request sent to OSS.

You can add the parameters in the following three methods:

- Add the parameters to the URL.
- Add the parameters to the header.
- Add the parameters to the form fields in the body of a POST request.



#### Note:

You can use only this method to specify callback parameters when uploading objects using POST requests.

The preceding three methods are alternative. If you use more than one method, OSS returns an `InvalidArgument` error.

To add the parameters to a request sent to OSS, you must use base64 to encode the JSON string constructed in the preceding section, and then add the parameters as follows:

- To add the parameters to the URL, add `callback =[ Callback ]` or `callback - var =[ CallbackVa r ]` to the request as a URL parameter. When the `CanonicalizedResource` field in the signature is calculated, `callback` or `callback-var` is used as a sub-resource.
- To add the parameters to the header, add `x - oss - callback =[ Callback ]` or `x - oss - callback - var =[ CallbackVa r ]` to the request as a header. When the `CanonicalizedOSSHeaders` field in the signature is calculated, `x-oss-callback-var` and `x-oss-callback` are used. The code example is as follows:

```
PUT / test . txt HTTP / 1 . 1
Host : callback - test . oss - test . aliyun - inc . com
```

```

Accept - ncoding : identity
Content - Length : 5
x - oss - callback - var : eyJ40m15X3 ZhciI6ImZv ciljYWxsYm
Fjay10ZXN0 In0 =
User - Agent : aliyun - sdk - python / 0 . 4 . 0 ( Linux / 2 . 6
. 32 - 220 . 23 . 2 . alii089 . el5 . x86_64 / x86_64 ; 2 . 5 . 4 )
x - oss - callback : eyJjYWxsYm Fja1VybCI6 IjEyMS40My
4xMTMuODoy MzQ1Ni9pbm RleC5odG1s IiwgICJjYW xsYmFja0Jv
ZHkiOiJidW NrZXQ9JHti dWNrZXR9Jm 9iamVjdD0k e29iamVjdH
0mZXRhZz0k e2V0YWd9Jn NpemU9JHtz aXplfSZtaW 1lVHlwZT0k
e2lpbWVUeX BlfSZpbWFn ZuluZm8uaG VpZ2h0PSR7 aW1hZ2VJbm
ZvLmhlwWdo dH0maW1hZ2 VJbmZvLndp ZHRoPSR7aW 1hZ2VJbmZv
LndpZHRofS ZpbWFnZulu Zm8uZm9ybW F0PSR7aW1h Z2VJbmZvLm
ZvcmlhdH0m bXlfdmFyPS R7eDpteV92 YXJ9In0 =
Host : callback - test . oss - test . aliyun - inc . com
Expect : 100 - Continue
Date : Mon , 14 Sep 2015 12 : 37 : 27 GMT
Content - Type : text / plain
Authorizat ion : OSS mlepou3zr4 u7b14 : 5a74vhd4UX
pmyuudV14K aen5cY4 =
Test

```

- Use form fields to add parameters to the body of a POST request.
- It is slightly complicated to add the callback parameter when the POST method is used to upload an object because the callback parameter must be added using an independent form field, as shown in the following example:

```

-- 9431149156 168
Content - Dispositio n : form - data ; name =" callback "
eyJjYWxsYm Fja1VybCI6 IjEwLjEwMS 4xNjYuMzA6 ODA4My9jYW
xsYmFjay5w aHAiLCJjYW xsYmFja0hv c3QiOiIxMC 4xMDEuMTY2
LjMwIiwY2 FsbGJhY2tC b2R5IjoiZm lsZW5hbWU9 JChmaWxlbm
FtZSkmdGFi bGU9JHt4On RhYmxlfSIs ImNhbGxiYW NrQm9keVR5
cGUiOiJhcH BsaWNhdGlv bi94LXd3dy 1mb3JtLXVy bGVuY29kZW QifQ
==

```

- Custom parameters cannot be added by including the callback-var parameter to a form field. Each custom parameter must be added by using an independent form field. For example, if the JSON string for the custom parameters is as follows:

```

{
  " x : var1 ":" value1 ",
  " x : var2 ":" value2 "
}

```

The form fields in the POST request are as follows:

```

-- 9431149156 168
Content - Dispositio n : form - data ; name =" callback "
eyJjYWxsYm Fja1VybCI6 IjEwLjEwMS 4xNjYuMzA6 ODA4My9jYW
xsYmFjay5w aHAiLCJjYW xsYmFja0hv c3QiOiIxMC 4xMDEuMTY2
LjMwIiwY2 FsbGJhY2tC b2R5IjoiZm lsZW5hbWU9 JChmaWxlbm
FtZSkmdGFi bGU9JHt4On RhYmxlfSIs ImNhbGxiYW NrQm9keVR5
cGUiOiJhcH BsaWNhdGlv bi94LXd3dy 1mb3JtLXVy bGVuY29kZW QifQ
==

```

```
-- 9431149156 168
Content - Dispositio n : form - data ; name =" x : var1 "
value1
-- 9431149156 168
Content - Dispositio n : form - data ; name =" x : var2 "
value2
```

You can also add callback conditions in the policy (if callback parameters are not added, upload verification is not performed on this parameter). For example :

```
{ " expiration ": " 2014 - 12 - 01T12 : 00 : 00 . 000Z ",
  " conditions ": [
    {" bucket ": " johnsmith " },
    {" callback ": " eyJjYWxsYm Fja1VybcI6 IjEwLjEwMS
4xNjYuMzA6 ODA4My9jYW xsYmFjay5w aHAiLCJjYW xsYmFja0hv
c3QiOiIxMC 4xMDEuMTY2 LjMwIiwIY2 FsbGJhY2tC b2R5IjoizM
lsZW5hbWU9 JChmaWxlbm FtZSkiLCJj YWxsYmFja0 JvZHLUeXBl
IjoiyXBwbG ljYXRpb24v eC13d3ctZm 9ybS11cmxl bmNvZGVkIn 0
="},
    [" starts - with ", "$ key ", " user / eric /"],
  ]
}
```

### Step 3: Initiate a callback request.

If a file is uploaded successfully, OSS sends the content specified by the callback and callback-var parameters in the request to the application server by using the POST method as follows:

```
POST / index . html HTTP / 1 . 0
Host : 121 . 43 . 113 . 8
Connection : close
Content - Length : 181
Content - Type : applicatio n / x - www - form - urlencoded
User - Agent : ehttp - client / 0 . 0 . 1
bucket = callback - test & object = test . txt & etag = D8E8FCA2DC
0F896FD7CB 4CB0031BA2 49 & size = 5 & mimeType = text % 2Fplain &
imageInfo . height =& imageInfo . width =& imageInfo . format =& x :
var1 = for - callback - test
```

### (Optional) Step 4: Sign the callback request.

If the callback parameter is configured in the request, OSS initiates a POST callback request to the application server through the URL specified by the callbackUrl field. To verify whether the callback request received by the application server is initiated by OSS, you can sign the callback request.

- Generate a signature.

A call request is signed by OSS using the RSA asymmetric algorithm.

A signature is generated by encrypting the callback string with a private key, as shown in the following code:

```
authorization = base64_encode ( rsa_sign ( private_key ,
url_decode ( path ) + query_string + '\n' + body , md5 ) )
```



**Note:**

In the preceding code, `private_key` is a private key only known by OSS, `path` is the resource path included in the callback request, `query_string` is the query string, and `body` is the message body specified in the callback request.

A callback request is signed in the following steps:

1. Obtain the callback string to be signed, which is composed by the resource path obtained by decoding the URL, the original query string, a carriage return, and the callback message body.
2. Sign the callback string with the RSA encryption algorithm, that is, using the private key to encrypt the callback string. The hash function used for signature is MD5.
3. Use Base64 to encode the signed result to get the final signature and Add the signature to the authorization header in the callback request.

The example of a signed request is as follows

```
POST / index . php ? id = 1 & index = 2 HTTP / 1 . 0
Host : 121 . 43 . 113 . 8
Connection : close
Content - Length : 18
authorization : kKQeGTRccD KyHB3H9vF + xYMSrmhMZj zzl2 /
kdD1ktNVgb WEfYTQG0G2 SU / RaHBovRCE8 OkQDjC3uG3 3esH2txA ==
Content - Type : applicatio n / x - www - form - urlencoded
User - Agent : ehttp - client / 0 . 0 . 1
x - oss - pub - key - url : aHR0cDovL2 dvc3NwdWJs aWMuYWxpY2
RuLmNvbS9j YWxsYmFja1 9wdWJfa2V5 X3YxLnBlbQ ==
bucket = yonghu - test
```

In the preceding example, `path` is `/ index . php` , `query_string` is `? id = 1 &`

`index = 2` , and `body` is `bucket = yonghu - test` . The final signature is

`kKQeGTRccD KyHB3H9vF + xYMSrmhMZj zzl2 / kdD1ktNVgb WEfYTQG0G2 SU`  
`/ RaHBovRCE8 OkQDjC3uG3 3esH2txA ==`.

- Verify the signature.

Signature verification is an inverse process of signing a request. The signature is verified by the application server as follows:

```
Result = rsa_verify ( public_key , md5 ( url_decode ( path ) +
query_string + '\ n ' + body ), base64_decode ( authorization ))
```

The fields in the preceding code have the same meanings as they are used to sign the request, in which `public_key` indicates the public key, `authorization` indicates the signature in the callback request header. The signature is verified as follows:

1. The `x-oss-pub-key-url` header in the callback request stores the base64-encoded URL of the public key. Therefore, you must decode the base64-coded URL to get the public key,

```
public_key = urlopen ( base64_decode ( x - oss - pub - key -
url_header ))
```



#### Note:

To ensure that the public key is issued by OSS, you must verify whether the value of the `x-oss-pub-key-url` header starts with `http://gosspublic.alicdn.com` / `or` `https://gosspublic.alicdn.com` /.

2. Obtain the decoded signature.

```
signature = base64_decode ( authorization_header )
```

3. Obtain the string to be signed the same way as described in the process of signing the callback request.

```
sign_str = url_decode ( path ) + query_string + '\ n ' +
body
```

4. Verify the signature.

```
result = rsa_verify ( public_key , md5 ( sign_str ),
signature )
```

The preceding sample is used as an example:

1. Obtain the URL of the public key by base64-decoding `aHR0cDovL2`

```
dvc3NwdWJs aWMuYWxpY2 RuLmNvbS9j YWxsYmFja1 9wdWJfa2V5
```

X3YxLnBlbQ ==. The decoded URL is `http :// gosspublic . alicdn . com / callback_p ub_key_v1 . pem .`

2. Base64-decode the signature header `kKQeGTRccD KyHB3H9vF + xYMSrmhMZj zzl2 / kdD1ktNVgb WEfYTQG0G2 SU / RaHBovRCE8 0kQDjC3uG3 3esH2txA ==`. (The decoded result cannot be displayed because it is a nonprintable string.)
  3. Obtain the string to be signed, that is, `url_decode( "index.php" ) + "?id=1&index=2" + "\n" + "bucket=yonghu-test"` , and perform the MD5 verification on the string.
  4. Verify the signature.
- Application server example

The following Python code demonstrates how an application server verifies a signature. To run the code, the M2Crypto library must be installed.

```
import httpplib
import base64
import md5
import urllib2
from BaseHTTPServer import BaseHTTPRequestHandler, HTTPServer
from M2Crypto import RSA
from M2Crypto import BIO
def get_local_ip():
    try:
        csock = socket.socket(socket.AF_INET, socket.SOCK_DGRAM)
        csock.connect(('8.8.8.8', 80))
        (addr, port) = csock.getsockname()
        csock.close()
        return addr
    except socket.error:
        return ""
class MyHTTPReqHandler(BaseHTTPRequestHandler):
    '''
    def log_message(self, format, *args):
        return
    '''
    def do_POST(self):
        # get public key
        pub_key_url = ''
        try:
            pub_key_url_base64 = self.headers['x-oss-pub-key-url']
            pub_key_url = pub_key_url_base64.decode('base64')
            if not pub_key_url.startswith("http://gosspublic.alicdn.com/") and not pub_key_url.startswith("https://gosspublic.alicdn.com/"):
                self.send_response(400)
                self.end_headers()
                return
            url_reader = urllib2.urlopen(pub_key_url)
```



```

        # you can cache it
        pub_key = url_reader.read()
    except:
        print 'pub_key_url: ' + pub_key_url
        print 'Get pub key failed!'
        self.send_response(400)
        self.end_headers()
        return
    # get authorization
    authorization_base64 = self.headers['authorization']
    authorization = authorization_base64.decode('base64')
    # get callback body
    content_length = self.headers['content-length']
    callback_body = self.rfile.read(int(content_length))
    # compose authorization string
    auth_str = ''
    pos = self.path.find('?')
    if -1 == pos:
        auth_str = urllib2.unquote(self.path) + '\n' + callback_body
    else:
        auth_str = urllib2.unquote(self.path[0:pos]) + self.path[pos:] + '\n' + callback_body
    print auth_str
    # verify authorization
    auth_md5 = md5.new(auth_str).digest()
    bio = BIO.MemoryBuffer(pub_key)
    rsa_pub = RSA.load_public_key_bio(bio)
    try:
        result = rsa_pub.verify(auth_md5, authorization, 'md5')
    except:
        result = False
    if not result:
        print 'Authorization verify failed!'
        print 'Public key: %s' % (pub_key)
        print 'Auth string: %s' % (auth_str)
        self.send_response(400)
        self.end_headers()
        return
    # do something according to callback_body
    # response to OSS
    resp_body = '{"Status": "OK"}'
    self.send_response(200)
    self.send_header('Content-Type', 'application/json')
    self.send_header('Content-Length', str(len(resp_body)))
    self.end_headers()
    self.wfile.write(resp_body)
class MyHTTPServer(HTTPServer):
    def __init__(self, host, port):
        HTTPServer.__init__(self, (host, port), MyHTTPRequestHandler)
if '__main__' == __name__:
    server_ip = get_local_ip()
    server_port = 23451
    server = MyHTTPServer(server_ip, server_port)

```

```
server . serve_fore ver ()
```

The code for the application server in other languages is as follows

Java:

- Click [here](#) to download the code.
- Running method: Extract the package and run `java -jar oss - callback - server - demo . jar 9000` (9000 is the port number and can be specified as needed).

PHP:

- Click [here](#) to download the code.
- Running method: Deploy the code to an Apache environment because some headers in the PHP code is depended on the environment. You can modify the example code according to the environment.

Python:

- Click [here](#) to download the code.
- Running method: Extract the package and run `python callback_a pp_server . py` . To run the code, RSA dependencies must be installed.

C#:

- Click [here](#) to download the code.
- Running method: Extract the package and see `README . md` to get more information.

.NET :

- Click [here](#) to download the code.
- Running method: Extract the package and see `README . md` to get more information.

Go:

- Click [here](#) to download the code.
- Running method: Extract the package and see `README . md` to get more information.

Ruby:

- Click [here](#) to download the code.
- Running method: Run ruby aliyun\_oss\_callback\_server.rb.

Step 5: Return the callback result.

The application server returns the response to OSS.

The response to the callback request is as follows:

```
HTTP / 1 . 0    200    OK
Server :   BaseHTTP / 0 . 3    Python / 2 . 7 . 6
Date :   Mon , 14    Sep    2015    12 : 37 : 27    GMT
Content - Type : applicatio n / json
Content - Length : 9
{" a ":" b "}
```



**Note:**

The response returned by the application server to OSS must contain the Content-Length header, and the size of the response body cannot exceed 1 MB.

Step 6: Return the upload result.

OSS returns the information returned by the application server to the user.

The returned response is as follows:

```
HTTP / 1 . 1    200    OK
Date :   Mon , 14    Sep    2015    12 : 37 : 27    GMT
Content - Type : applicatio n / json
Content - Length : 9
Connection : keep - alive
ETag : " D8E8FCA2DC 0F896FD7CB 4CB0031BA2 49 "
Server : AliyunOSS
x - oss - bucket - version : 1442231779
x - oss - request - id : 55F6BF8720 7FB30F2640 C548
{" a ":" b "}
```



**Note:**

- The body of responses for some requests (such as CompleteMultipartUpload) contains content (for example, information in XML format). If you use the upload callback function, the original body content is overwritten, such as {" a ":" b "}. Take this into consideration when you use the upload callback function.
- If the upload callback fails, a 203 error is returned with the error code CallbackFailed. This indicates that the file is successfully uploaded to OSS but the callback fails. A callback failure only indicates that OSS does not receive the expected callback response, but not indicates that the application server does not receive a

callback request. For example, the response returned by the application server is not in JSON format.

## 4.7 PutSymlink


PutSymlink is used to create a symbolic link directing to the TargetObject on OSS. You can use the symbolic link to access the TargetObject.

### Request syntax

```
PUT / ObjectName ? symlink HTTP / 1 . 1
Host : BucketName . oss - cn - hangzhou . aliyuncs . com
Date : GMT Date
Authorization : SignatureValue
x - oss - symlink - target : TargetObjectName
```

### Request header

| Name                       | Type   | Description  |
|----------------------------|--------|--|
| x - oss - symlink - target | String | Indicates the target object that a symbolic link directs to.<br>Valid value: The naming conventions are the same as those for objects. |

| Name                         | Type   | Description  |
|------------------------------|--------|--|
| x - oss - storage<br>- class | String | <p>Specifies the storage class of the target object.<br/><b>Values:</b></p> <ul style="list-style-type: none"> <li>• Standard</li> <li>• IA</li> <li>• Archive</li> </ul> <p>Supported interfaces: PutObject, InitMultiPartUpload, AppendObject, PutObjectSymlink, and CopyObject</p> <div>  <b>Note:</b> <ul style="list-style-type: none"> <li>• If the value of StorageClass is invalid, a 400 error is returned. Error code: InvalidArgument</li> <li>• We recommend that you do not set the storage class in PutObjectSymlink to IA or Archive because an IA or Archive object smaller than 64 KB is billed at 64 KB.</li> <li>• If you specify the value of x-oss-storage-class when uploading an object to a bucket, the storage class of the uploaded object is the specified value of x-oss-storage-class regardless of the storage class of the bucket. For example, if you specify the value of x-oss-storage-class to Standard when uploading an object to a bucket of the IA storage class, the storage class of the object is Standard.</li> </ul> </div> |

#### Detail analysis

- Similar to ObjectName, TargetObjectName must be URL-encoded.
- The target object that a symbolic link directs to cannot be a symbolic link.
- When a symbolic link is created, the following checks are not performed:
  - Whether the target object exists.
  - Whether the storage class of the target object is valid.
  - Whether the user has permission to access the target object.

These checks are performed by APIs that access the target object, such as GetObject.

- If the object that you want to add already exists and you can access the object, the existing object is overwritten by the added object and a 200 OK message is returned.
- If a PutSymlink request carries a parameter with the x-oss-meta- prefix, the parameter is considered as user meta, such as x-oss-meta-location. An object can have multiple parameters with the x-oss-meta- prefix. However, the total size of all user meta cannot exceed 8 KB.

### Example

#### Request example:

```
PUT / link - to - oss . jpg ? symlink HTTP / 1 . 1
Host : oss - example . oss - cn - hangzhou . aliyuncs . com
Cache - control : no - cache
Content - Dispositio n : attachment ; filename = oss_downlo ad .
jpg
Date : Tue , 08 Nov 2016 02 : 00 : 25 GMT
Authorizat ion : OSS qn6qrrqxo2 oawuk53otf jbyc : kZoYNv66bs
mc10 + dcGKw5x2PR rk = x - oss - symlink - target : oss . jpg
x - oss - storage - class : Standard
```

#### Response example:

```
HTTP / 1 . 1 200 OK
Server : AliyunOSS
Date : Tue , 08 Nov 2016 02 : 00 : 25 GMT
Content - Length : 0
Connection : keep - alive
x - oss - request - id : 582131B910 9F4EE66CDE 56A5
ETag : " 0A477B89B4 602AA8DECB 8E19BFD447 B6 "
```

## 4.8 SelectObject

SelectObject is used to query an object that you have the read permission on.

### SelectObject

SelectObject is used to run SQL statements on the target object and return the query result.

The 206 status code is returned if the operation is successfully performed. If the SQL statements are incorrect or do not match the target object, the 400 status code is returned.



#### Note:

For more information about the functions of SelectObject, see [SelectObject](#).

- Request syntax

- Request syntax (CSV)

```
POST / object ? x - oss - process = csv / select HTTP / 1 . 1

HOST : BucketName . oss - cn - hangzhou . aliyuncs . com
Date : time GMT
Content - Length : ContentLength
Content - MD5 : MD5Value
Authorization : Signature

<? xml version =" 1 . 0 " encoding =" UTF - 8 ">
< SelectRequest >
  < Expression > base64 encode ( Select * from OSSObject
where ...)</ Expression >
  < InputSerialization >
    < CompressionType > None | GZIP </ CompressionType >
    < CSV >
      < FileHeaderInfo >
        NONE | IGNORE | USE
      </ FileHeaderInfo >
      < RecordDelimiter > base64 encode </ RecordDelimiter >
      < FieldDelimiter > base64 encode </ FieldDelimiter >
      < QuoteCharacter > base64 encode </ QuoteCharacter >
      < CommentCharacter > base64 encode </ CommentCharacter >
      < Range > line - range = start - end | split - range = start -
end </ Range >
    </ CSV >
  </ InputSerialization >
  < OutputSerialization >
    < CSV >
      < RecordDelimiter > base64 encode </ RecordDelimiter >
      < FieldDelimiter > base64 encode </ FieldDelimiter >

    </ CSV >
    < KeepAllColumns > false | true </ KeepAllColumns >
    < OutputRawData > false | true </ OutputRawData >
      < EnablePayloadCrc > true </ EnablePayloadCrc >
    < OutputHeader > false </ OutputHeader >
  </ OutputSerialization >
  < Options >
    < SkipPartialDataRecord > false </ SkipPartialDataRecord >
    < MaxSkippedRecordsAllowed >
      max allowed number of records skipped
    < MaxSkippedRecordsAllowed >
  </ Options >
</ SelectRequest >
```

- Request syntax (JSON)

```
POST / object ? x - oss - process = json / select HTTP / 1 . 1

HOST : BucketName . oss - cn - hangzhou . aliyuncs . com
Date : time GMT
Content - Length : ContentLength
Content - MD5 : MD5Value
Authorization : Signature

<? xml version =" 1 . 0 " encoding =" UTF - 8 ">
< SelectRequest >
```

```

< Expression >
  Base64 encode of sql such as ( select * from
ossobject )
</ Expression >
< InputSerialization >
  < CompressionType > None | GZIP </ CompressionType >
  < JSON >
    < Type > DOCUMENT | LINES </ Type >
    < Range >
      line - range = start - end | split - range = start - end
    </ Range >
    < ParseJsonNumberAsString > true | false
    </ ParseJsonNumberAsString >
  </ JSON >
</ InputSerialization >
< OutputSerialization >
  < JSON >
    < RecordDelimiter >
      Base64 of record delimiter
    </ RecordDelimiter >
  </ JSON >
  < OutputRawData > false | true </ OutputRawData >
  < EnablePayloadCrc > true </ EnablePayloadCrc >
</ OutputSerialization >
< Options >
  < SkipPartialDataRecord >
    false | true
  </ SkipPartialDataRecord >
  < MaxSkippedRecordsAllowed >
    max allowed number of records skipped
    < MaxSkippedRecordsAllowed >
  </ Options >
</ SelectRequest >

```

• Request elements

| Element            | Type      | Description   |
|--------------------|-----------|---|
| SelectRequest      | Container | Specifies the container that saves the SelectObject request.<br>Child nodes: Expression, InputSerialization, and OutputSerialization<br>Parent node: None |
| Expression         | String    | Specifies the base64-coded SQL statements.<br>Child node: None<br>Parent node: SelectRequest  |
| InputSerialization | Container | (Optional) Specifies the input serialization parameters.<br>Child nodes: CompressionType, CSV, and JSON<br>Parent node: SelectRequest                     |




| Element                  | Type                    | Description   |
|--------------------------|-------------------------|---|
| OutputSerialization      | Container               | (Optional) Specifies the output serialization parameters.<br>Child nodes: CSV, JSON, and OutputRawData<br>Parent node: SelectRequest  |
| CSV(InputSerialization)  | Container               | (Optional) Specifies the format parameter for the input CSV file.<br>Child nodes: FileHeaderInfo, RecordDelimiter, FieldDelimiter, QuoteCharacter, CommentCharacter, and Range<br>Parent node: InputSerialization |
| CSV(OutputSerialization) | Container               | (Optional) Specifies the format parameter for the output CSV file.<br>Child nodes: RecordDelimiter and FieldDelimiter<br>Parent node: OutputSerialization   |
| JSON(InputSerialization) | Container               | Specifies the format parameter for the input JSON file.<br>Child node: Type   |
| Type                     | Enumeration             | Specifies the type of the input JSON file:<br>DOCUMENT   LINES  |
| JSON(InputSerialization) | Container               | Specifies the format parameter for the input JSON file.<br>Child node: RecordDelimiter  |
| OutputRawData            | Bool (false by default) | (Optional) Specifies the output data as raw data, which is not the frame-based format.<br>Child node: None<br>Parent node: OutputSerialization  |
| CompressionType          | Enumeration             | Specifies the compression type of the object: None GZIP<br>Child node: None<br>Parent node: InputSerialization  |

| Element         | Type        | Description  |
|-----------------|-------------|--|
| FileHeaderInfo  | Enumeration | <p>(Optional) Specifies the header information about the CSV file.</p> <p>Valid values:</p> <ul style="list-style-type: none"> <li>- <b>Use:</b> Indicates that the CSV file contains header information. You can use the column name in the CSV file as the column name in the SelectObject operation.</li> <li>- <b>Ignore:</b> Indicates that the CSV file contains header information. However, you cannot use the column name in the CSV file as the column name in the SelectObject operation.</li> <li>- <b>None:</b> Indicates that the CSV file does not contain header information. This is the default value.</li> </ul> <p>Child node: None<br/>Parent node: CSV (input)</p> |
| RecordDelimiter | String      | <p>(Optional) Specifies the delimiter, which is base64-encoded and <code>\ n</code> by default. The value of this element before being encoded can be the ANSI value of two characters in maximum. For example, <code>\ n</code> is used to indicate a line break in Java code.</p> <p>Child node: None<br/>Parent node: CSV (input and output) and JSON (output)</p>  |
| FieldDelimiter  | String      | <p>(Optional) Specifies the delimiter used to separate columns in the CSV file. The value of this element is the base64-encoded ANSI value of a character and is <code>,</code> by default. For example, <code>,</code> is used to indicate a comma in Java code.</p> <p>Child node: None<br/>Parent node: CSV (input and output)</p>  |

| Element          | Type   | Description   |
|------------------|--------|---|
| QuoteCharacter   | String | <p>(Optional) Specifies the quote characters used in the CSV file. The value of this element is base64-encoded and is <code>\</code> by default. In a CSV file, line breaks and column delimiters are processed as normal characters. The value of this element before being encoded must be the ANSI value of a character. For example, <code>\</code> is used to indicate a quote character in Java code.</p> <p>Child node: None<br/>Parent node: CSV (input)</p>  |
| CommentCharacter | String | <p>Specifies the comment character used in the CSV file. The value of this element is base64-encoded and is null (no comment character) by default.</p>   |
| Range            | String | <p>(Optional) Specifies the query range. The following two query methods are supported:</p> <ul style="list-style-type: none"> <li>- Query by rows: line-range=start-end</li> <li>- Query by splits:split-range=start-end</li> </ul> <p>The start and end parameters in the preceding code are both inclusive. The format of the two parameters are the same as that of the range parameter in range get operations.</p> <p>This parameter is valid only when the document is in CSV format or the JSON Type is LINES.</p> <p>Child node: None<br/>Parent nodes: CSV (input) and JSON (input)</p> |

| Element          | Type      | Description   |
|------------------|-----------|---|
| KeepAllColumns   | Bool      | <p>(Optional) Indicates that all columns in the CSV file are included in the returned result. However, only columns included in the select statement have values. The default value of this parameter is false. The columns in the returned result are sorted in order of the column numbers from low to high. For example:</p> <pre>select _5 , _1 from ossobject .</pre> <p>If you set the value of KeepAllColumn to true and six columns are included in the CSV file, the following result is returned for the preceding select statement:</p> <p>Value of 1st column,,,,Value of 5th column,\n</p> <p>Child node: None</p> <p>Parent node: OutputSerialization (CSV)</p> |
| EnablePayloadCrc | Bool      | <p>Indicates that each frame includes a 32-bit CRC32 value for verification. The client can calculate the CRC32 value of each payload and compare it with the included CRC32 value to verify data integrity.</p> <p>Child node: None</p> <p>Parent node: OutputSerialization</p>  |
| Options          | Container | <p>Specifies other optional parameters.</p> <p>Child node: SkipPartialDataRecord and MaxSkippedRecordsAllowed</p> <p>Parent node: SelectRequest</p>   |
| OutputHeader     | Bool      | <p>Indicates that the header information about the CSV file is included in the beginning of the returned result.</p> <p>Default value: false</p> <p>Child node: None</p> <p>Parent node: OutputSerialization</p>  |

| Element                  | Type    | Description  |
|--------------------------|---------|--|
| SkipPartialDataRecord    | Bool    | <p>Indicates that rows without data are ignored. If the value of this parameter is false, OSS ignores rows without data (by processing the values of the rows as null) and does not report errors. If the value of this parameter is true, a row without data is skipped. If the number of skipped rows exceeds the maximum allowed number, OSS reports an error and stops processing the data.</p> <p>Default value: false<br/>Child node: None<br/>Parent node: Options</p>  |
| MaxSkippedRecordsAllowed | Integer | <p>Specifies the maximum allowed number of skipped rows. If a row does not match the type specified in the SQL statement, or a column or multiple columns in a row are missed and the value of SkipPartialDataRecord is True, the row is skipped. If the number of skipped rows exceeds the value of this parameter, OSS reports an error and stops processing the data.</p> <div>  <p><b>Note:</b><br/>If a row is not in the valid CSV format, for example, a column in the row includes continual odd numbered quote characters, OSS stops processing the data immediately and reports an error because this format error may result in incorrect resolution to the CSV file. That is, this parameter can be used to adjust the tolerance for irregular data but does not applied to invalid CSV files.</p> </div> <p>Default value: 0<br/>Child node: None<br/>Parent node: Options</p> |

| Element                 | Type | Description   |
|-------------------------|------|---|
| ParseJsonNumberAsString | Bool | Indicates that the numbers (integer and float numbers) in a JSON file are resolved into strings. The accuracy of float numbers in a JSON file degrades when the numbers are resolved. Therefore, we recommend that you set the value of this parameter to true if you want to keep the original data. To use the numbers for calculation, you can cast them into the required format, such as int, double, or decimal, in the SQL statement.<br>Default value: false<br>Child node: None<br>Parent node: JSON |

- Response body

If the HTTP status included in the response for a request is 4xx, it indicates that the request does not pass the SQL syntax check or an obvious error is included in the request. In this case, the body format of the returned error message is the same as that of the error message returned for a GetObject request.

If the HTTP status code included in the response for a request is 5xx, it indicates that an error occurs in the server. In this case, the body format of the returned message is the same as that of the error message returned for a GetObject request.

If the HTTP status code 206 is returned in the response and the value of header x-oss-select-output-raw is true, it indicates that the object data (but not frame-based data) is successfully returned. The client can obtain the data in the same way as that used in GetObject operations.

If the value of x-oss-select-output-raw is false, the result is returned as frames.

If you set a value for OutputRawData in a request, OSS returns the requested data in the format that you specified. However, we recommend that you do not set a

value for `OutputRawData` so that OSS returns the requested data in the format automatically select by OSS.

If you set the value of `OutputRawData` to `true` in an HTTP request, the request may be time out when no data is returned for the SQL statement for a long period.

If you perform a `SelectObject` operation using a JSON file and the select statement includes repeated keys (for example: `select s.key, s.key from ossoobject s`), the value of the `x-oss-select-output-json-dup-key` header in the response is `true`.

A returned frame is in the following format, in which the checksum is CRC32

Version|Frame-Type | Payload Length | Header Checksum | Payload | Payload Checksum

<1 byte><--3 bytes--><---4 bytes----><-----4 bytes--><variable><----4bytes----->

All integers in a frame are big-endian. Currently, the value of Version is 1.

`SelectObject` supports three frame types, as described in the following table.

| Frame type       | Frame -Type value | Payload format                            | Description  |
|------------------|-------------------|---|--|
| Data Frame       | 8388609           | offset   data<br><-8 bytes><---variable-> | A data frame includes the data returned for the <code>SelectObject</code> request. The offset parameter is an 8-bit integer, which indicates the current scanning location (the offset from the file header ) and is used to report the progress of the operation. |
| Continuous Frame | 8388612           | offset<br><----8 bytes-->                 | A continuous frame is used to report the progress of an operation and keep an HTTP connection. If no data is returned for a query request within 5 seconds, a continuous frame is returned.  |

| Frame type | Frame<br>-Type<br>value | Payload format   | Description   |
|------------|-------------------------|--|---|
| End Frame  | 8388613                 | offset   total<br>scanned bytes  <br>http status code  <br>error message<br><--8bytes-><--<br>8bytes-----><----<br>4 bytes-----><-<br>variable-----> | <p>An end frame is used to return the final status of an operation, including the scanned bytes and the final offset. The total scanned bytes parameter indicates the size of the scanned data, the http status code parameter indicates the final status of the operation, and the error message parameter includes error messages, including the number of each skipped row and the total number of skipped rows.</p> <p>SelectObject is a streamed operation so that only the first data block is processed when the response header is sent. If the first data block matches the SQL statement, the HTTP status code in the response header is 206, which indicates that the operation is successful. However, the final status code may not be 206 because the following data blocks may be valid but the status code in the response header cannot be modified in this case. Therefore the HTTP status code is included in the end frame to indicate the final status of the operation. The client should use the status code included in the end frame to determine whether the operation is successful.</p> |



- Error messages

The format of error messages included in an end frame is as follows:

```
ErrorCodes . DetailMessage
```

The ErrorCodes part includes a single ErrorCode or multiple ErrorCodes separated by commas. The ErrorCodes and DetailMessage part are separated by a period. For detailed error codes, see the ErrorCode list at the end of this topic.

- Example requests

- Example request (CSV)

```
POST / oss - select / bigcsv_nor mal . csv ? x - oss - process
= csv % 2Fselect HTTP / 1 . 1
Date : Fri , 25 May 2018 22 : 11 : 39 GMT
Content - Type :
Authorization : OSS LTAIJPXxML ocA0fD : FC / 9JRbBGRw4o
2QqdaL246P xuvk =
User - Agent : aliyun - sdk - dotnet / 2 . 8 . 0 . 0 ( windows
16 . 7 / 16 . 7 . 0 . 0 / x86 ; 4 . 0 . 30319 . 42000 )
Content - Length : 748
Expect : 100 - continue
Connection : keep - alive
Host : host name

<? xml version =" 1 . 0 ">
< SelectRequest >
  < Expression > c2VsZWN0IG NvdW50KCop IGZyb20gb3 Nzb2JqZWN0
  IHdoZXJlIF 80ID4gNDU =
  </ Expression >
  < InputSerialization >
    < Compression > None </ Compression >
    < CSV >
      < FileHeader Info > Ignore </ FileHeader Info >
      < RecordDelimiter > Cg ==</ RecordDelimiter >
      < FieldDelimiter > LA ==</ FieldDelimiter >
      < QuoteCharacter > Ig ==</ QuoteCharacter >
      < CommentCharacter > Iw ==</ CommentCharacter />
    </ CSV >
  </ InputSerialization >
  < OutputSerialization >
    < CSV >
      < RecordDelimiter > Cg ==</ RecordDelimiter >
      < FieldDelimiter > LA ==</ FieldDelimiter >
      < QuoteCharacter > Ig ==</ QuoteCharacter >
    </ CSV >
    < KeepAllColumns > false </ KeepAllColumns >
    < OutputRawData > false </ OutputRawData >
  </ OutputSerialization >
</ SelectRequest >
```

- Example request (JSON)

```
POST / oss - select / sample_json . json ? x - oss - process =
json % 2Fselect HTTP / 1 . 1
Host : host name
Accept - Encoding : identity
```

```

User - Agent : aliyun - sdk - python / 2 . 6 . 0 ( Darwin / 16 .
7 . 0 / x86_64 ; 3 . 5 . 4 )
Accept : */*
Connection : keep - alive
date : Mon , 10 Dec 2018 18 : 28 : 11 GMT
authorizat ion : OSS AccessKeyS ignature
Content - Length : 317

< SelectRequ est >
< Expression > c2VsZWN0IC ogZnJvbSBv c3NvYmplY3 Qub2JqZWN0
c1sqXSB3aG VyZSBwYXJ0 eSA9ICdEZW 1vY3JhdCc =
</ Expression >
< InputSeria lization >
< JSON >
  < Type > DOCUMENT </ Type >
</ JSON >
</ InputSeria lization >
< OutputSeri alization >
< JSON >
  < RecordDeli miter > LA ==</ RecordDeli miter >
</ JSON >
</ OutputSeri alization >
< Options />
</ SelectRequ est >

```

- **Regular expressions in an SQL statement**

```
SELECT select - list from table where_opt limit_opt
```

**SELECT, OSSOBJECT, and WHERE are keywords that cannot be modified.**

```

select_lis t : column name
| column index ( for example : _1 , _2 . column index
only applies to CSV files )
| json path ( for example : s . contacts . firstname . json
path only applies to JSON files )
| function ( column index | column name )
| function ( json_path ) ( only applies for JSON files )

```

```
| select_list AS alias
```

The following functions are supported: AVG, SUM, MAX, MIN, COUNT, and CAST (type conversion function). You can use only the wildcard (\*) after COUNT.

table: OSSOBJECT

| OSSOBJECT json\_path (only supported for JSON files)

For a CSV file, the table must be OSSOBJECT. For a JSON file (including DOCUMENT and LINES types), you can specify a json\_path after OSSOBJECT.

json\_path: ['string'] (The brackets can be deleted if the string does not include a space or a wildcard (\*), that is, 'string'.)

| [n] (Used to indicate the nth element in an array. The value of n is counted from 0.)

| [\*] (Used to indicate any child element in an array or object.)

| 'string' (The quotation marks around string can be deleted if the string does not include a space or a wildcard (\*).)

| json\_path jsonpath (You can concatenate multiple elements in a json path, for example, [n].property1.attributes[\*].)

```
Where_opt :
| WHERE expr
expr :
| literal value
| column name
| column index
| json path ( only applies to JSON files )
| expr op expr
| expr OR expr
| expr AND expr
| expr IS NULL
| expr IS NOT NULL
| ( column name | column index | json path ) IN ( value1
, value2 ,...)
| ( column name | column index | json path ) NOT in (
value1 , value2 ,...)
| ( column name | column index | json path ) between
value1 and value2
| NOT ( expr )
| expr op expr
| ( expr )
```

```
| cast ( column index | column name | json path |
| literal as INT | DOUBLE |)
```

**op:** includes the following operators: >, <, >=, <=, !=, =, LIKE, +, -, \*, /, %, and ||.

**cast:** You can only cast the data in a same column to one type.

**limit\_opt :**

| limit integer

Combination use of an aggregation function and limit

```
Select avg ( cast ( _1 as int )) from ossobject limit
100
```

The preceding statement calculates the average values of the first columns in the first 100 rows, which is different from the MySQL statement. It is because only one row is returned for a aggregation function in SelectObject operations so that it is unnecessary to limit its output. Therefore, limit is performed before aggregation functions in SelectObject operations.

Limits for SQL statements

- Only text files encoded in UTF-8 and UTF-8 text files compressed in the GZIP format are supported. The deflate format is not supported for GZIP files.
- An SQL statement can query only one file. The following commands are not supported: join, order by, group by, and having.
- A Where statement cannot include an aggregation condition. For example, the following statement is not allowed: where max(cast(age as int)) > 100.
- A maximum of 1,000 columns are supported. The maximum column number is 1024.
- A maximum of 5 wildcard "%" are supported in a LIKE statement. The wildcard "%" plays the same role as the wildcard "\*", which is used to indicate 0 or multiple characters. The keyword Escape is supported in a LIKE statement,

which is used to escape the special characters (such as "%", "\*", and "?") into normal strings.

- A maximum of 1,024 constants are supported in an IN statement.
- The Projection after Select can be a column name, a CSV column index (such as \_1 and \_2), an aggregation function, or a CAST function. Other expressions are not supported, for example, `select _1 + _2 from ossobject`.
- The maximum column size and row size for a CSV file are 256 KB.
- The json path after from supports a JSON node with a maximum size of 512 KB. The path can have 10 levels at most and includes a maximum of 5,000 elements in the array.
- In SQL statements for a JSON file, the select or where expressions cannot include the array wildcard ([\*]), which can be included only in the json path after from. For example, `select s . contacts [*] from ossobject s` is not supported but `select * from ossobject . contacts [*]` is supported.
- The maximum size of an SQL statement is 16 KB. A maximum of 20 expressions can be added after where. A statement supports at most 10 levels and 100 aggregation operations.
- Data error handling
  - Some columns are missed in some rows in a CSV file.

If the value of `SkipPartialDataRecord` is not specified or is set to false, OSS calculates the expressions in the SQL statement by processing the values of the missed columns as null.

If the value of `SkipPartialDataRecord` is set to true, OSS ignores the rows in which some columns are missed. In this case, if the value of `MaxSkippedRecordsAllowed` is not specified or is set to a value smaller than the number

of skipped rows, OSS returns an error by sending the 400 HTTP status code or including the 400 status code in the end frame.

For example, assuming that the SQL statement is `select _1 , _3 from ossobject` and the data in a row of the CSV file is "John, company A".

If the value of `SkipPartialDataRecord` is set to false, the returned result is "John, \n". If the value of `SkipPartialDataRecord` is set to true, this row is ignored.

- Some keys are missed in a JSON file.

Some objects in the JSON file may not include the keys specified in the SQL statement. In this case, if the value of `SkipPartialDataRecord` is set to false, OSS calculates the expressions in the SQL statement by processing the missed keys as null.

If the value of `SkipPartialDataRecord` is true, OSS ignores the data in the JSON node. In this case, if the value of `MaxSkippedRecordsAllowed` is not specified or is set to a value smaller than the number of ignored rows, OSS returns an error by sending the 400 HTTP status code or including the 400 status code in the end frame.

For example, assuming that the SQL statement is `select s . firstName , s . lastName , s . age from ossobject . contacts [*] s` and the value of a JSON node is { "firstName" : " John" , " lastName" : " Smith" }.

If the value of `SkipPartialDataRecord` is not specified or be set to false, the returned result is { "firstName" : " John" , "lastName" : " Smith" }. If the value of `SkipPartialDataRecord` is set to true, this row is ignored.

- The data type of some columns in row in a CSV file does not match the SQL statement.

If the data type of some rows in a CSV file does not match the type specified in the SQL statement, this row is ignored. If the number of ignored rows exceeds

the value of `MaxSkippedRecordsAllowed`, OSS stops processing data and returns a 400 HTTP status code.

For example, assuming that the SQL statement is `select _1 , _3 from ossobject where _3 > 5`.

If the value of a row in the CSV file is `John , Company A , To be hired`, this row is ignored because the third column in the row is not an integer.

- The data type of some keys in a JSON file does not match the SQL statement.

The handling method is the same as that in a CSV file. For example, assuming that the SQL statement is `select s . name from ossobject s where s . aliren_age > 5`.

If the value of a JSON node is `{"Name":"John", "Career age":To be hired}`, this node is ignored.

- Keys in a returned JSON file.

The returned result for a `SelectObject` operation using a JSON file is a file in the JSON LINES format, in which the keys are determined as follows:

- If the SQL statement is `select * from ossobject ...`, if a JSON object (`{...}`) is returned for the wildcard (`*`), the object is directly returned. If the returned result is not a JSON object (for example, a string or an array), a `DummyKey_1` is used to indicate the returned result.

For example, if the data is `{ "Age" :5}` and the SQL statement is `select * from ossobject . Age s where s = 5`. The result returned for the wildcard (`*`) is 5, which is not a JSON object. Therefore, the returned result for the statement is `{ "_1" :5}`. However, if the statement is `select * from ossobject s where s . Age = 5`, the result returned for the

wildcard (\*) is the JSON object { "Age" :5}, so that the object is directly returned for the statement.

- If the SQL statement does not use `select *` but specifies a column, the format of the response is as follows: `{"{ column1 }": value , "{ column2 }": value ...}`.

In the response, the value of "n" in {column n} is generated as follows:

- If the alias of the column is specified in the SelectObject request, the value of n is set to the specified alias.
- If the column is a key of a JSON object, the key is used as the output key.
- If the column is an aggregation function or an element in a JSON array, the serial number of the column in the output result followed by a prefix `_` is used as the key of the output result.

For example, if the data is { "contacts" :{ "Age" :35, "Children" :[ "child1" , "child2" , " child3" ]}}, and the SQL statement is `select s . contacts . Age , s . contacts . Children [ 0 ] from ossobjects` , the output result is { "Age" :35, "\_2" : " child1" }. This result is returned because Age is a key of the input JSON object, but Children [0] is the first element in the array Children, which is in the second column in the output result.

- If the alias of the row is specified in the request, the output result for `select s . contacts . Age , s . contacts . Children [ 0 ] as firstChild from ossobject` is { "Age" :35, "firstChild" : " child1" }.
- If the SQL statement is `select max ( cast ( s . Age as int )) from ossobject . contacts s` , the output result is { "\_1" :35}, in which the serial number of the column with the prefix `_1` is used to indicate the key because this row is a aggregation function.



#### Note:

Keys in a JSON file are case-sensitive when they are used to match the keys in an SQL statement. For example, `select s_Age` and `select s_age` are different keys.



## CreateSelectObjectMeta

CreateSelectObjectMeta is used to obtain information about the target CSV file, such as the total number of rows, the total number of columns, and the number of Splits. If the information does not exist in the file, the whole CSV file is scanned for the preceding information. The information obtained in the first call of the API is used when the API is called again, so that you do not need to scan the whole CSV file. If the API is executed correctly, the 200 status code is returned. If the target file is not a valid CSV or JSON LINES file, or the specified delimiter does not match the target CSV file, the 400 HTTP status code is returned.



### Note:

You must have the write permission on the target object before performing a CreateSelectObjectMeta operation.

- Request syntax
  - Request syntax (CSV)

```
POST    / samplecsv ? x - oss - process = csv / meta

< CsvMetaReq uest >
< InputSeria lization >
  < Compressio nType > None </ Compressio nType >
  < CSV >
    < RecordDeli miter > base64   encode </ RecordDeli miter >
    < FieldDelim iter > base64   encode </ FieldDelim iter >
    < QuoteChara cter > base64   encode </ QuoteChara cter >
  </ CSV >
</ InputSeria lization >
< OverwriteI fExists > false | true </ OverwriteI fExists >
</ CsvMetaReq uest >
```

- Request syntax (JSON)

```
POST    / samplecsv ? x - oss - process = json / meta

< JsonMetaRe quest >
< InputSeria lization >
  < Compressio nType > None </ Compressio nType >
  < JSON >
    < Type > LINES </ Type >
  </ JSON >
</ InputSeria lization >
< OverwriteI fExists > false | true </ OverwriteI fExists >
```

```
</ JsonMetaRequest >
```

• Request elements

| Element            | Type        | Description  |
|--------------------|-------------|--|
| CsvMetaRequest     | Container   | Specifies the container that saves the Select csv Meta request.<br>Child node: InputSerialization<br>Parent node: None   |
| JsonMetaRequest    | Container   | Specifies the container that saves the Select json Meta request.<br>Child node: InputSerialization<br>Parent node: None  |
| InputSerialization | Container   | (Optional) Specifies the input serialization parameters.<br>Child node: CompressionType, CSV, and JSON<br>Parent node: CsvMetaRequest and JsonMetaRequest  |
| OverwriteIfExists  | Bool        | (Optional) Recalculates the SelectMeta and overwrites the existing data. The default value is false, which means that the result is directly returned if the Select Meta already exists.<br>Child node: None<br>Parent node: CsvMetaRequest and JsonMetaRequest                          |
| CompressionType    | Enumeration | (Optional) Specifies the compression type of the object. Only None is supported currently.<br>Child node: None<br>Parent node: InputSerialization  |
| RecordDelimiter    | String      | (Optional) Specifies the delimiter, which is base64-encoded and \n by default. The value of this element before being encoded can be the ANSI value of two characters in maximum. For example, \n is used to indicate a line break in Java code.<br>Child node: None<br>Parent node: CSV |

| Element        | Type        | Description  |
|----------------|-------------|--|
| FieldDelimiter | String      | (Optional) Specifies the delimiter used to separate columns in the CSV file. The value of this element is the base64-encoded ANSI value of a character and is <code>,</code> by default. For example, <code>,</code> is used to indicate a comma in Java code.<br>Child node: None<br>Parent node: CSV (input and output)  |
| QuoteCharacter | String      | (Optional) Specifies the quote characters used in the CSV file. The value of this element is base64-encoded and is <code>\</code> by default. In a CSV file, line breaks and column delimiters are processed as normal characters. The value of this element before being encoded must be the ANSI value of a character. For example, <code>\</code> is used to indicate a quote character in Java code.<br>Child node: None<br>Parent node: CSV (input) |
| CSV            | Container   | Specifies the format of the input CSV file.<br>Child node: RecordDelimiter, FieldDelimiter, and QuoteCharacter<br>Parent node: InputSerialization  |
| JSON           | Container   | Specifies the format of the input JSON file.<br>Child node: Type<br>Parent node: InputSerialization  |
| Type           | Enumeration | Specifies the type of the input JSON file.<br>Valid value: LINES   |

Similar to `SelectObject`, the results for `CreateSelectObjectMeta` is also returned as frames, which have two types: continuous frames and end meta frames.

Continuous frames used for CreateSelectObjectMeta is the same as those used for SelectObject.

| Frame type              | Frame<br>-Type<br>value | Payload format   | Description   |
|-------------------------|-------------------------|--|---|
| Meta End<br>Frame (CSV) | 8388614                 | offset   status  splits<br>count   rows count  <br>columns count   error<br>message<br><-8 bytes><--4bytes><--4<br>bytes--><--8 bytes><--4<br>bytes---><variable size> | offset: A 8-bit integer that indicates the offset when the scanning is complete.<br>status: A 4-bit integer that indicates the final status of the operation.<br>splits_count: A 4-bit integer that indicates the number of splits.<br>rows_count: A 8-bit integer that indicates the total number of rows.<br>cols_count: A 4-bit integer that indicates the total number of columns.<br>error_message: Includes detailed error messages. If no error occurs, the value of this parameter is null.<br>Meta End Frame: Used to report the final status of a CreateSelectObjectMeta operation. |

| Frame type            | Frame-Type value | Payload format   | Description  |
|-----------------------|------------------|--|--|
| Meta End Frame (JSON) | 8388615          | offset   status   splits count   rows count   error message<br><-8 bytes><--4bytes><--4 bytes--><--8 bytes><variable size> | offset: A 8-bit integer that indicates the offset when the scanning is complete.<br>status: A 4-bit integer that indicates the final status of the operation.<br>splits_count: A 4-bit integer that indicates the number of splits.<br>rows_count: A 8-bit integer that indicates the total number of rows.<br>error_message: Includes detailed error messages. If no error occurs, the value of this parameter is null.<br>Meta End Frame: Used to report the final status of a CreateSelectObjectMeta operation. |

**Response Header:** No specified header is included in the response.

- **Example requests**

- **Example request (CSV)**

```
POST / oss - select / bigcsv_nor mal . csv ? x - oss - process
= csv % 2Fmeta HTTP / 1 . 1
Date : Fri , 25 May 2018 23 : 06 : 41 GMT
Content - Type :
Authorizat ion : OSS AccessKeyS ignature
User - Agent : aliyun - sdk - dotnet / 2 . 8 . 0 . 0 ( windows
16 . 7 / 16 . 7 . 0 . 0 / x86 ; 4 . 0 . 30319 . 42000 )
Content - Length : 309
Expect : 100 - continue
Connection : keep - alive
Host : Host

<? xml version =" 1 . 0 ">
< CsvMetaReq uest >
< InputSeria lization >
< CSV >
< RecordDeli miter > Cg ==</ RecordDeli miter >
< FieldDelim iter > LA ==</ FieldDelim iter >
< QuoteChara cter > Ig ==</ QuoteChara cter >
</ CSV >
</ InputSeria lization >
```

```
< OverwriteI fExisting > false </ OverwriteI fExisting >
</ CsvMetaReq uest >
```

- Example request (JSON)

```
POST / oss - select / sample . json ? x - oss - process = json %
2Fmeta HTTP / 1 . 1
Date : Fri , 25 May 2018 23 : 06 : 41 GMT
Content - Type :
Authorizat ion : OSS AccessKeyS ignature
User - Agent : aliyun - sdk - dotnet / 2 . 8 . 0 . 0 ( windows
16 . 7 / 16 . 7 . 0 . 0 / x86 ; 4 . 0 . 30319 . 42000 )
Content - Length : 309
Expect : 100 - continue
Connection : keep - alive
Host : Host

<? xml version =" 1 . 0 ">
< JsonMetaRe quest >
< InputSeria lization >
< JSON >
< Type > LINES </ Type >
</ JSON >
</ InputSeria lization >
< OverwriteI fExisting > false </ OverwriteI fExisting >
</ JsonMetaRe quest >
```

### Supported time format

You can transfer a string in the formats listed in the following table into a timestamp without specifying the time format. For example, the string cast('20121201' as timestamp) is automatically resolved into a timestamp: 1st, December, 2012.

The following table describes the time formats that can be automatically recognized and transferred.

| Format       | Description                    |
|--------------|--------------------------------|
| YYYYMMDD     | year month day                 |
| YYYY/MM/DD   | year/month/day                 |
| DD/MM/YYYY/  | day/month/year                 |
| YYYY-MM-DD   | year-month-day                 |
| DD-MM-YY     | day-month-year                 |
| DD.MM.YY     | day.month.year                 |
| HH:MM:SS.mss | hour:minute:second.millisecond |
| HH:MM:SS     | hour:minute:second             |
| HH MM SS mss | hour minute second millisecond |
| HH.MM.SS.mss | hour.minute.second.millisecond |

| Format                  | Description   |
|-------------------------|---|
| HHMM                    | hour second   |
| HHMMSSmss               | hour minute second millisecond  |
| YYYYMMDD HH:MM:SS.mss   | year month day hour:minute:second.<br>millisecond   |
| YYYY/MM/DD HH:MM:SS.mss | year/month/day hour:minute:second.<br>millisecond   |
| DD/MM/YYYY HH:MM:SS.mss | day/month/year hour:minute:second.<br>millisecond   |
| YYYYMMDD HH:MM:SS       | year month day hour:minute:second   |
| YYYY/MM/DD HH:MM:SS     | year/month/day hour:minute:second   |
| DD/MM/YYYY HH:MM:SS     | day/month/year hour:minute:second   |
| YYYY-MM-DD HH:MM:SS.mss | year-month-day hour:minute:second.<br>millisecond   |
| DD-MM-YYYY HH:MM:SS.mss | day-month-year hour:minute:second.<br>millisecond   |
| YYYY-MM-DD HH:MM:SS     | year-month-day hour:minute:second   |
| YYYYMMDDTHH:MM:SS       | year month day T hour:minute:second   |
| YYYYMMDDTHH:MM:SS.mss   | year month day T hour:minute:second.<br>millisecond   |
| DD-MM-YYYYTHH:MM:SS.mss | day-month-year T hour:minute:second.<br>millisecond   |
| DD-MM-YYYYTHH:MM:SS     | day-month-year T hour:minute:second   |
| YYYYMMDDTHHMM           | year month day T hour minute  |
| YYYYMMDDTHHMMSS         | year month day T hour minute second   |
| YYYYMMDDTHHMMSSMSS      | year month day T hour minute second<br>millisecond  |
| ISO8601-0               | year-month-day T hour:minute+hour<br>:minute, or year-month-day T hour:<br>minute-hour:minute<br>"+" indicates that the time in the current<br>timezone is in front of standard UTC time<br>"."-" indicates that the time in the current<br>timezone is behind the stand UTC time.<br>In this format, ISO8601-0 can be used to<br>indicate "+". |

| Format            | Description  |
|-------------------|--|
| ISO8601-1         | year-month-day T hour:minute+hour:minute, or year-month-day T hour:minute-hour:minute<br>"+" indicates that the time in the current timezone is in front of standard UTC time<br>"."-" indicates that the time in the current timezone is behind the stand UTC time.<br>In this format, ISO8601-1 can be used. |
| CommonLog         | Such as 28/Feb/2017:12:30:51 +0700   |
| RFC822            | Such as Tue, 28 Feb 2017 12:30:51 GMT  |
| ?D/?M/YY          | day/month/year, in which the day and month can be a 1-bit or 2-bit number.   |
| ?D/?M/YY ?H:?M    | day month year hour:minute, in which the day, month, hour, and minute can be a 1-bit or 2-bit number.  |
| ?D/?M/YY ?H:?M:?S | day month year hour:minute:second, in which the day, month, hour, minute, and second can be a 1-bit or 2-bit number.   |

The formats in the following table are ambiguous. You must specify a time format when using strings in these formats. For example, the cast('20121201' as timestamp format 'YYYYDDMM') statement incorrectly resolves the string 20121201 to 12nd, January, 2012.

| Format     | Description    |
|------------|----------------|
| YYYYDDMM   | year day month |
| YYYY/DD/MM | year/day/month |
| MM/DD/YYYY | month/day/year |
| YYYY-DD-MM | year-day-month |
| MM-DD-YYYY | month-day-year |
| MM.DD.YYYY | month.day.year |

## ErrorCode

SelectObject returns Errorcodes in the following two methods:

- Include the HTTP status code in the response headers and include error messages in the response body, which is the same as other OSS requests. An ErrorCode



returned in this way indicates that an obvious input or data error (such as an invalid SQL statement is input) occurs.

- Include the Error code in the end frame of the response body. An ErrorCode returned in this way indicates that the data is not correct or does not match the SQL statement. For example, a string exists in a column of which the type is specified as integer in the SQL statement. In this case, a part of data is processed and returned to the client, and the status code is 206.

Some ErrorCode (such as InvalidCSVLine) can be returned as the HTTP status code in the response header or the status code included in the end frame according to the location of the error row in the CSV file.

| ErrorCode                   | Description  | HTTP status code | Http status code in end frame |
|-----------------------------|--|------------------|-------------------------------|
| InvalidSqlParameter         | Invalid SQL parameter.<br>Indicates that the SQL statement in the request is null, the SQL statement size exceeds the limit, or the SQL statement is not base64-encoded. | 400              | None                          |
| InvalidInputFieldDelimiter  | Invalid column delimiter in the input CSV file.<br>Indicates that the parameter is not base64-encoded or the size of the parameter is larger than 1 after being decoded. | 400              | None                          |
| InvalidInputRecordDelimiter | Invalid row delimiter in the input CSV file. Indicates that the parameter is not base64-encoded or the size of the parameter is larger than 2 after being decoded.       | 400              | None                          |
| InvalidInputQuote           | Invalid quote character in the input CSV file. Indicates that the parameter is not base64-encoded or the size of the parameter is larger than 1 after being decoded.     | 400              | None                          |

| ErrorCode                       | Description  | HTTP status code | Http status code in end frame |
|---------------------------------|--|------------------|-------------------------------|
| InvalidOutputFieldDelimiter     | Invalid column delimiter in the output CSV file. Indicates that the parameter is not base64-encoded or the size of the parameter is larger than 1 after being decoded.                                 | 400              | None                          |
| InvalidOutputRecordDelimiter    | Invalid row delimiter in the output CSV file. Indicates that the parameter is not base64-encoded or the size of the parameter is larger than 2 after being decoded.                                    | 400              | None                          |
| UnsupportedCompressionFormat    | Invalid Compression parameter. Indicates that the value of the parameter is not NONE or GZIP (case-insensitive).   | 400              | None                          |
| InvalidCommentCharacter         | Invalid comment character in the CSV file. Indicates that the parameter is not base64-encoded or the size of the parameter is larger than 1 after being decoded.                                       | 400              | None                          |
| InvalidRange                    | Invalid Range parameter. Indicates that the parameter is not prefixed with <code>line - range =</code> or <code>split - range =</code> , or the range value does not meet the HTTP standard for Range. | 400              | None                          |
| DecompressFailure               | Indicates that the value of Compression is GZIP and the decompression fails.   | 400              | None                          |
| InvalidMaxSkippedRecordsAllowed | Indicates that the value of MaxSkippedRecordsAllowed is not an integer.  | 400              | None                          |
| SelectCsvMetaUnavailable        | Indicates that CreateSelectObjectMeta is firstly called when the Range parameter is specified but the target object does not include CSV Meta.   | 400              | None                          |
| InvalidTextEncoding             | Indicates that the object is not UTF-8 encoded.  | 400              | None                          |

| ErrorCode                        | Description   | HTTP status code | Http status code in end frame |
|----------------------------------|---|------------------|-------------------------------|
| InvalidOSSSelectParameters       | Indicates the EnablePayloadCrc and OutputRawData parameters are both set to true, which results in conflicts. | 400              | None                          |
| InternalError                    | Indicates that an OSS system error occurs.  | 500 or 206       | None or 500                   |
| SqlSyntaxError                   | Indicates that the syntax of the base64-decoded SQL statement is incorrect.                                   | 400              | None                          |
| SqlExceedsMaxInCount             | Indicates that the number of values included in the IN statement exceeds 1,024.                               | 400              | None                          |
| SqlExceedsMaxColumnNameLength    | Indicates that the size of the column name exceeds 1,024.   | 400              | None                          |
| SqlInvalidColumnIndex            | Indicates that the column index in the SQL statement is smaller than 1 or larger than 100.                    | 400              | None                          |
| SqlAggregationOnNonNumericType   | Indicates that an aggregation function is used in a non-numeric column.                                       | 400              | None                          |
| SqlInvalidAggregationOnTimestamp | Indicates that the SUM/AVG aggregation function is used in the timestamp column.                              | 400              | None                          |
| SqlValueTypeOfInMustBeSame       | Indicates that values of different types are included in the IN statement.                                    | 400              | None                          |
| SqlInvalidEscapeChar             | Indicates that escape characters in the LIKE statement is "?", "%", or "*".                                   | 400              | None                          |
| SqlOnlyOneEscapeCharIsAllowed    | Indicates that the size of the escape character in the LIKE statement is larger than 1.                       | 400              | None                          |
| SqlNoCharAfterEscapeChar         | Indicates that there are no character after the escape character in the LIKE statement.                       | 400              | None                          |

| ErrorCode                           | Description  | HTTP status code | Http status code in end frame |
|-------------------------------------|--|------------------|-------------------------------|
| SqlInvalidLimitValue                | Indicates that the number after the Limit statement is smaller than 1.   | 400              | None                          |
| SqlExceedsMaxWildCardCount          | Indicates that the number of wildcards ("*" or "%") exceeds the limit in the LIKE statement.   | 400              | None                          |
| SqlExceedsMaxConditionCount         | Indicates that the number of conditional expressions in the Where statement exceeds the limit.   | 400              | None                          |
| SqlExceedsMaxConditionDepth         | Indicates that the depth of the conditional tree in the Where statement exceeds the limit.   | 400              | None                          |
| SqlOneColumnCastToDifferentTypes    | Indicates that a column is casted to different types in the SQL statement.   | 400              | None                          |
| SqlOperationAppliedToDifferentTypes | Indicates that an operator is used for two objects of different type in the SQL statement. For example, this ErrorCode is returned if the col1 in <code>_col1 &gt; 3</code> is a string. | 400              | None                          |
| SqlInvalidColumnName                | Indicates that a column name used in the SQL statement is not included in the header of the CSV file.  | 400              | None                          |
| SqlNotSupportedTimestampFormat      | Indicates that the timestamp format specified in the CAST statement is not supported.  | 400              | None                          |
| SqlNotMatchTimestampFormat          | Indicates that the timestamp format specified in the CAST statement does not match the timestamp string.   | 400              | None                          |
| SqlInvalidTimestampValue            | Indicates that no timestamp format is specified in the CAST statement and the provided timestamp string cannot be casted into a timestamp.   | 400              | None                          |

| ErrorCode                                   | Description   | HTTP status code | Http status code in end frame |
|---|---|------------------|-------------------------------|
| SqlInvalidLikeOperand                       | Indicates that the left column in the LIKE statement is not column names of column indexes, the left column in the LIKE statement is not the string type, or the right column in the LIKE statement is the string type. | 400              | None                          |
| SqlInvalidMixOfAggregationAndColumn         | Indicates that the SQL statement includes the column names and indexes of aggregation functions and non-aggregation functions at the same time.   | 400              | None                          |
| SqlExceedsMaxAggregationCount               | Indicates that the number of aggregation functions included in the SQL statement exceeds the limit.   | 400              | None                          |
| SqlInvalidMixOfStarAndColumn                | Indicates that the wildcard "*", column name, and column index are included in the SQL statement at the same time.  | 400              | None                          |
| SqlInvalidKeepAllColumnsWithAggregation     | Indicates that the SQL statement includes aggregation functions while the KeepAllColumns parameter is set to true.  | 400              | None                          |
| SqlInvalidKeepAllColumnsWithDuplicateColumn | Indicates that the SQL statement include repeated column names or indexes while the KeepAllColumns parameter is set to true.  | 400              | None                          |
| SqlInvalidSqlAfterAnalysis                  | Indicates that the SQL statement is not supported because it is too complicated after being resolved.   | 400              | None                          |
| InvalidArithmeticOperand                    | Indicates that arithmetical operations are performed on non-numeric constants or columns in the SQL statement.  | 400              | None                          |
| SqlInvalidAndOperand                        | Indicates that the type of expressions connected by AND in the SQL statement is not bool.   | 400              | None                          |

| ErrorCode                     | Description   | HTTP status code | Http status code in end frame |
|-------------------------------|---|------------------|-------------------------------|
| SqlInvalidOrOperand           | Indicates that the type of expressions connected by OR in the SQL statement is not bool.  | 400              | None                          |
| SqlInvalidNotOperand          | Indicates that the type of expressions connected by NOT in the SQL statement is not bool.   | 400              | None                          |
| SqlInvalidIsNullOperand       | Indicates that the IS NULL operation is performed on a constant in the SQL statement.   | 400              | None                          |
| SqlCompareOperandTypeMismatch | Indicates that the SQL statement compares two objects of different types.   | 400              | None                          |
| SqlInvalidConcatOperand       | Indicates that two constants are connected by the string connect operator (  ) in the SQL statement.  | 400              | None                          |
| SqlUnsupportedSql             | Indicates that the SQL statement is too complicated so that the size of the generated SQL plan exceeds the limit.   | 400              | None                          |
| HeaderInfoExceedsMaxSize      | Indicates that the size of header information specified in the SQL statement exceeds the limit.   | 400              | None                          |
| OutputExceedsMaxSize          | Indicates that a single row of output results exceeds the limit size.   | 400              | None                          |
| InvalidCsvLine                | Indicates that a row in the CSV file is invalid (including that the size of the row exceeds the limit) or the number of ignored rows exceeds the value of MaxSkippedRecordsAllowed. | 206 or 400       | 400 or none                   |
| NegativeRowIndex              | Indicates that the value of the array Index in the SQL statement is a negative number.  | 400              | None                          |

| ErrorCode                        | Description  | HTTP status code | Http status code in end frame |
|----------------------------------|--|------------------|-------------------------------|
| ExceedsMaxNestedColumnDepth      | Indicates that the nested levels of the JSON file in the SQL statement exceeds the limit.  | 400              | None                          |
| NestedColumnNotSupportedInCsv    | Indicates that the nested attributes (including array "[]" or ".") are not supported in the CSV file in the SQL statement.   | 400              | None                          |
| TableRootNodeOnlySupportedInJson | Indicates that the root node path can be specified after From ossobject only in JSON files.  | 400              | None                          |
| JsonNodeExceedsMaxSize           | Indicates that the size of the root node in the JSON file exceeds the limit.   | 400 or 206       | None or 400                   |
| InvalidJsonData                  | Indicates that the JSON data is invalid (incorrect format).  | 400 or 206       | None or 400                   |
| ExceedsMaxJsonArraySize          | Indicates that the number of elements in an array in the root node of the JSON file exceeds the limit.   | 400 or 206       | None or 400                   |
| WildcardNotAllowed               | Indicates that the wildcard (*) in the cannot be used in the JSON file in select or where statements. For example, the following statement is not supported:<br><pre>select  s . a . b [*] from ossobject  where  a . c [*] &gt; 0 .</pre> | 400              | None                          |
| JsonNodeExceedsMaxDepth          | Indicates that the depth of the root node of the JSON file exceeds the limit.  | 400 or 206       | None or 400                   |

## 5 Multipart upload operations

### 5.1 InitiateMultipartUpload

Before transmitting data in Multipart Upload mode, you must call the `InitiateMultipartUpload` interface to require OSS to initiate a Multipart Upload event.

The `InitiateMultipartUpload` interface returns a globally unique upload ID created by the OSS server to identify this Multipart Upload event. You can initiate operations based on this Upload ID, such as stopping or querying the Multipart Upload.

#### Request syntax

```
POST / ObjectName ? uploads HTTP / 1 . 1
Host : BucketName . oss - cn - hangzhou . aliyuncs . com
Date : GMT date
Authorizat ion : SignatureV alue
```

#### Request parameters


During the `InitiateMultipartUpload` operation, you can use the encoding-type to encode the Key in the returned result.

| Name            | Type   | Description   |
|-----------------|--------|---|
| encoding - type | String | Specifies the encoding type of the Key in the returned result. Currently, URL encoding is supported. The Key is UTF-8-encoded, but the XML 1.0 standard does not support parsing certain control characters, such as the characters with ASCII values from 0 to 10. If the Key contains control characters not supported by the XML 1.0 standard, you can specify the encoding-type to encode the returned Key.<br>Default value: None<br>Optional value: url |



## Request header

| Name                                 | Type    | Description   |
|--------------------------------------|---------|---|
| Cache - control                      | String  | Specifies the Web page caching behavior when the object is downloaded. For more information, see <a href="#">RFC2616</a> .<br>Default value: None   |
| Content - Dispositio n               | String  | Specifies the object name when the object is downloaded. For more information, see <a href="#">RFC2616</a> .<br>Default value: None   |
| Content - Encoding                   | String  | Specifies the content encoding format when the object is downloaded. For more information, see <a href="#">RFC2616</a> .<br>Default value: None   |
| Expires                              | Integer | Specifies the expiration time in milliseconds. For more information, see <a href="#">RFC2616</a> .<br>Default value: None   |
| x - oss - server - side - encryption | String  | Specifies the server-side encryption algorithm used to upload each part of the object. OSS stores each uploaded part based on server-side encryption.<br>Valid value: AES256 or KMS<br>You must enable KMS (Key Management Service) in the console before you can use the KMS encryption algorithm. Otherwise, a KmsServiceNotenabled error code is reported. |

| Name                      | Type   | Description   |
|---------------------------|--------|---|
| x - oss - storage - class | String | <p>Specifies the storage class of the object.</p> <p><b>Values:</b></p> <ul style="list-style-type: none"> <li>Standard</li> <li>IA</li> <li>Archive</li> </ul> <p><b>Supported interfaces:</b> PutObject, InitMultipartUpload, AppendObject, PutObjectSymlink, and CopyObject</p> <div>  <b>Note:</b> <ul style="list-style-type: none"> <li>If the value of StorageClass is invalid, a 400 error is returned. Error code: InvalidArgument</li> <li>If you specify the value of x-oss-storage-class when uploading an object to a bucket, the storage class of the uploaded object is the specified value of x-oss-storage-class regardless of the storage class of the bucket. For example, if you set the value of x-oss-storage-class to Standard when uploading an object to a bucket of the IA storage class, the storage class of the object is Standard.</li> </ul> </div> |

## Response elements

| Name                          | Type      | Description   |
|-------------------------------|-----------|---|
| Bucket                        | String    | <p>Indicates the name of a bucket for which a Multipart Upload event is initiated.</p> <p>Parent element: InitiateMultipartUploadResult</p>                           |
| InitiateMultipartUploadResult | Container | <p>Indicates the container that saves the result of the InitiateMultipartUpload request.</p> <p>Child elements: Bucket, Key, UploadId</p> <p>Parent element: None</p> |

| Name         | Type   | Description   |
|--------------|--------|---|
| Key          | String | Indicates the name of an object for which a Multipart Upload event is initiated.<br>Parent element: InitiateMultipartUploadResult   |
| UploadId     | String | Indicates the unique ID of a Multipart Upload event.<br>Parent element: InitiateMultipartUploadResult   |
| EncodingType | String | Specifies the encoding type for the returned results. If the encoding-type parameter is specified in the request, the Key is encoded in the returned result.<br>Parent element: Container |

#### Detail analysis

- When performing this operation to calculate the authentication signature, you must add “?uploads” to CanonicalizedResource.
- InitiateMultipartUpload requests support the following standard HTTP request headers: Cache-Control, Content-Disposition, Content-Encoding, Content-Type, Expires, and custom headers starting with `x-oss-meta-`. For more information, see [PutObject](#).
- An InitiateMultipartUpload request does not affect the existing object with the same name.
- When receiving an InitiateMultipartUpload request, the server returns a request body in XML format. The request body includes three elements: Bucket, Key, and UploadID. You must record the UploadID for subsequent Multipart operations.
- If the x-oss-server-side-encryption header is set in the InitiateMultipartUpload request, the server returns this header in the response header. During the upload of each part, the server automatically stores the part based on entropy encryption. Currently, the OSS server only supports the AES256 and KMS encryption methods. If other methods are specified, the OSS server returns a 400 error with the InvalidEncryptionAlgorithmError error code. When uploading each part, you do not need to add the x-oss-server-side-encryption request header. If this request header is specified, OSS returns a 400 error with the InvalidArgument error code.

## Example

### Request example:

```
POST / multipart . data ? uploads HTTP / 1 . 1
Host : oss - example . oss - cn - hangzhou . aliyuncs . com
Date : Wed , 22 Feb 2012 08 : 32 : 21 GMT
x - oss - storage - class : Archive
Authorization : OSS qn6qrrqxo2 oawuk53otf jbyc :/ cluRFtRwMT
ZpC2hTj4F6 7AGdM4 =
```

### Response example:

```
HTTP / 1 . 1 200 OK
Content - Length : 230
Server : AliyunOSS
Connection : keep - alive
x - oss - request - id : 42c25703 - 7503 - fbd8 - 670a - bda01eaec6
18
Date : Wed , 22 Feb 2012 08 : 32 : 21 GMT
Content - Type : application / xml
<? xml version =" 1 . 0 " encoding =" UTF - 8 "? >
< InitiateMultipartUploadResult xmlns =" http :// doc . oss - cn
- hangzhou . aliyuncs . com ">
  < Bucket > multipart_ upload </ Bucket >
  < Key > multipart . data </ Key >
  < UploadId > 0004B9894A 22E5B1888A 1E29F8236E 2D </ UploadId >
</ InitiateMultipartUploadResult >
```

## 5.2 UploadPart

After initiating a Multipart Upload event, you can upload data in parts based on the specified object name and Upload ID. Each uploaded part has a part number ranging from 1 to 10,000.

For the same Upload ID, this part number identifies not only this part of data but also the location of this part in the entire file. If you upload new data using the same part number, OSS overwrites the existing data identified by this part number. The number of parts ranges from 1 to 10,000. The size of a single part ranges from 100 KB to 5 GB, while the last part can be less than 100 KB.

### Request syntax

```
PUT / ObjectName ? partNumber = PartNumber & uploadId = UploadId
HTTP / 1 . 1
Host : BucketName . oss - cn - hangzhou . aliyuncs . com
Date : GMT Date
Content - Length : Size
```

```
Authorization : SignatureV alue
```

### Detail analysis

- Before calling the Initiate Multipart Upload interface to upload a part of data, you must call this interface to obtain an Upload ID issued by the OSS server.
- In the Multipart Upload mode, except the last part, all other parts must be larger than 100 KB. However, the Upload Part interface does not immediately verify the size of the uploaded part (because it does not know whether the part is the last one ). It verifies the size of the uploaded part only when Multipart Upload is completed.
- OSS puts the MD5 value of the part data received by the server in the ETag header and return it to the user.
- The part number ranges from 1 to 10,000. If the part number exceeds this range, OSS returns the InvalidArgument error code.
- If the x-oss-server-side-encryption request header is specified when the Initiate Multipart Upload interface is called, OSS encrypts the uploaded part and return the x-oss-server-side-encryption header in the Upload Part response header. The value of x-oss-server-side-encryption indicates the server-side encryption algorithm used for this part.
- To make sure that the data transmitted over the network is free from errors, the user includes Content-MD5 in the request. The OSS calculates the MD5 value for the uploaded data and compares it with the MD5 value uploaded by the user. If they are inconsistent, OSS returns the InvalidDigest error code.

### Examples

#### Request example:

```
PUT / multipart . data ? partNumber = 1 & uploadId = 0004B9895D
BBB6EC98E3 6 HTTP / 1 . 1
Host : oss - example . oss - cn - hangzhou . aliyuncs . com
Content - Length : 6291456
Date : Wed , 22 Feb 2012 08 : 32 : 21 GMT
Authorization : OSS qn6qrrqxo2 oawuk53otf jbyc : J /
lICfXEvPmm SW86bBAfMm UmWjI =
[ 6291456 bytes data ]
```

#### Response example:

```
HTTP / 1 . 1 200 OK
Server : AliyunOSS
Connection : keep - alive
ETag : 7265F4D211 B56873A381 D321F586E4 A9
x - oss - request - id : 3e6aba62 - 1eae - d246 - 6118 - 8ff42cd0c2
1a
```

Date : Wed , 22 Feb 2012 08 : 32 : 21 GMT

## 5.3 UploadPartCopy

UploadPartCopy uploads a part by copying data from an existing object.

You can add an `x-oss-copy-source` header in the Upload Part request to call the Upload Part Copy interface. When copying a file larger than 1 GB, you must use the Upload Part Copy method. For the Upload Part Copy operation, the source bucket and the target bucket must be in the same region. If you want to copy a file that is less than 1 GB by a single operation, you can use the Copy Object method.

### Request syntax

```
PUT /ObjectName? partNumber = PartNumber & uploadId = UploadId
HTTP / 1.1
Host : BucketName . oss - cn - hangzhou . aliyuncs . com
Date : GMT Date
Content - Length : Size
Authorization : SignatureValue
x - oss - copy - source : / SourceBucketName / SourceObjectName
x - oss - copy - source - range : bytes = first - last
```

### Request header

Except the common request header, other headers in the Upload Part Copy request are used to specify the address of the copied source object and copying range.

| Name   | Type    | Description   |
|--|---------|---|
| <code>x - oss - copy - source</code>         | String  | Specifies the copy source address (the requester must have the permission to read the source object). Default: None   |
| <code>x - oss - copy - source - range</code> | Integer | Copying range of the copied source object. For example, if the range is set to bytes = 0-9, the system transfers byte 0 to byte 9. This request header is not required when the entire source object is copied. Default: None |

The following request header is used for the source objects specified by x-oss-copy-source.

| Name  | Type   | Description   |
|---|--------|---|
| x - oss - copy -<br>source - if - match                 | String | If the ETag value of the source object is equal to the ETag value provided by the user, the system performs the Copy Object operation; otherwise, the system returns the 412 Precondition Failed error.<br>Default: None                                      |
| x - oss - copy -<br>source - if - none -<br>match       | String | If the source object has not been modified since the time specified by the user, the system performs the Copy Object operation; otherwise, the system returns the 412 Precondition Failed error.<br>Default: None   |
| x - oss - copy<br>- source - if -<br>unmodified - since | String | If the time specified by the received parameter is the same as or later than the modification time of the file, the system transfers the file normally, and returns 200 OK; otherwise, the system returns the 412 Precondition Failed error.<br>Default: None |

| Name  | Type   | Description   |
|---|--------|---|
| x - oss - copy -<br>source - if - modified<br>- since | String | If the source object has been modified since the time specified by the user, the system performs the Copy Object operation; otherwise, the system returns the 412 Precondition Failed error.<br>Default: None |

## Response elements

| Name  | Type   | Description   |
|---|--------|---|
| x - oss - copy -<br>source - if - match                 | String | If the ETag value of the source object is equal to the ETag value provided by the user, the system performs the Copy Object operation; otherwise, the system returns the 412 Precondition Failed error.<br>Default: None                                      |
| x - oss - copy -<br>source - if - none -<br>match       | String | If the source object has not been modified since the time specified by the user, the system performs the Copy Object operation; otherwise, the system returns the 412 Precondition Failed error.<br>Default: None   |
| x - oss - copy<br>- source - if -<br>unmodified - since | String | If the time specified by the received parameter is the same as or later than the modification time of the file, the system transfers the file normally, and returns 200 OK; otherwise, the system returns the 412 Precondition Failed error.<br>Default: None |



| Name  | Type   | Description  |
|---|--------|--|
| x - oss - copy -<br>source - if - modified<br>- since | String | If the source object has been modified since the time specified by the user, the system performs the Copy Object operation; otherwise, the system returns the 412 Precondition Failed error. Default: None |

### Detail analysis

- Before calling the InitiateMultipartUpload interface to upload a part of data, you must call this interface to obtain an Upload ID issued by the OSS server.
- In the Multipart Upload mode, besides the last part, all other parts must be larger than 100 KB. However, the Upload Part interface does not immediately verify the size of the uploaded part (because it cannot immediately determine which part is the last one). It verifies the size of the uploaded part only when Multipart Upload is completed.
- If the x-oss-copy-source-range request header is not specified, the entire source object is copied. If the request header is specified, the returned message includes the length of the entire file and the COPY range. For example, if the returned message is Content-Range: bytes 0-9/44, which means that the length of the entire file is 44, and the COPY range is 0 to 9. If the specified range does not conform to the range rules, OSS copies the entire file and does not contain Content-Range in the result.
- If the x-oss-server-side-encryption request header is specified when the InitiateMultipartUpload interface is called, OSS encrypts the uploaded part and return the x-oss-server-side-encryption header in the Upload Part response header. The value of x-oss-server-side-encryption indicates the server-side encryption algorithm used for this part. For more information, see the InitiateMultipartUpload API.
- This operation cannot be used to copy objects created by Append Object.
- If the bucket type is Archive, you cannot call this interface; otherwise, the system returns Error 400 with the error code “OperationNotSupported” .

## Example

### Request example:

```
PUT / multipart . data ? Partnumber = 1 & sealadid =
porterhttp / 1 . 1
Host : oss - example . oss - cn - hangzhou . aliyuncs . com
Content - Length : 6291456
Date : Wed , 22 Feb 2012 08 : 32 : 21 GMT
Authorizat ion : OSS qn6qrrqxo2 oawuk53otf jbyc : J /
lICfXEvPmm SW86bBAfMm UmWjI =
x - oss - copy - source : / oss - example / src - object
x - oss - copy - source - range : bytes = 100 - 6291756
```

### Response example:

```
HTTP / 1 . 1 200 OK
Server : AliyunOSS
Connection : keep - alive
x - oss - request - id : 3e6aba62 - 1eae - d246 - 6118 - 8ff42cd0c2
1a
Date : Thu , 17 Jul 2014 06 : 27 : 54 GMT '
<? xml version =" 1 . 0 " encoding =" UTF - 8 " ? >
< CopyPartRe sult xmlns =" http :// doc . oss - cn - hangzhou .
aliyuncs . com ">
  < LastModifi ed > 2014 - 07 - 17T06 : 27 : 54 . 000Z </
LastModifi ed >
  < ETag >" 5B3C1A2E05 3D763E1B00 2CC607C5A0 FE "</ ETag >
</ CopyPartRe sult >
```