# Table of Contents

Amazon S3 REST API Introduction .................................................................................................................. 1  
Amazon S3 API Reference ................................................................................................................................. 2  
  Actions .......................................................................................................................................................... 2  
    Amazon Simple Storage Service ................................................................................................................. 5  
    AWS S3 Control ..................................................................................................................................... 408  
    Amazon S3 on Outposts ............................................................................................................................ 498  
Data Types .................................................................................................................................................... 506  
    Amazon Simple Storage Service .............................................................................................................. 510  
    AWS S3 Control ..................................................................................................................................... 676  
    Amazon S3 on Outposts ............................................................................................................................ 729  
Authenticating Requests (AWS Signature Version 4) .................................................................................... 733  
  Authentication Methods ............................................................................................................................... 734  
  Introduction to Signing Requests ................................................................................................................ 734  
  Using an Authorization Header .................................................................................................................... 735  
    Overview .............................................................................................................................................. 735  
    Signature Calculation: Transfer Payload in a Single Chunk .................................................................. 737  
    Signature Calculation: Transfer Payload in Multiple Chunks ............................................................... 748  
Using Query Parameters .................................................................................................................................. 755  
  Calculating a Signature .............................................................................................................................. 757  
  An Example .............................................................................................................................................. 759  
  Example 2 .............................................................................................................................................. 760  
Examples: Signature Calculations ................................................................................................................... 760  
  Signature Calculation Examples Using Java ............................................................................................... 761  
  Signature Calculation Examples Using C# .............................................................................................. 762  
Authenticating HTTP POST Requests .......................................................................................................... 762  
  Calculating a Signature .............................................................................................................................. 764  
  Amazon S3 Signature Version 4 Authentication Specific Policy Keys .................................................. 765  
  Bucket Policy Examples Using Signature Version 4 Related Condition Keys ........................................ 766  
Browser-Based Uploads Using POST ............................................................................................................ 768  
  POST Object ............................................................................................................................................ 769  
    Description ........................................................................................................................................... 769  
    Versioning ......................................................................................................................................... 769  
    Requests .......................................................................................................................................... 769  
    Examples .......................................................................................................................................... 779  
    Related Resources ............................................................................................................................... 779  
  POST Object restore ................................................................................................................................. 781  
    Description ........................................................................................................................................... 781  
    Querying Archives with Select Requests ............................................................................................. 781  
    Restoring Archives ............................................................................................................................... 782  
    Requests .......................................................................................................................................... 783  
    Responses ....................................................................................................................................... 792  
    Examples .......................................................................................................................................... 793  
    More Info ........................................................................................................................................... 795  
Browser-Based Uploads Using HTTP POST .................................................................................................... 795  
Calculating a Signature ................................................................................................................................. 796  
Creating HTML Forms ..................................................................................................................................... 797  
  HTML Form Declaration .......................................................................................................................... 797  
  HTML Form Fields ................................................................................................................................. 798  
Creating a POST Policy .................................................................................................................................. 801  
  Expiration .......................................................................................................................................... 802  
  Condition Matching ............................................................................................................................... 802  
  Conditions ...................................................................................................................................... 803  
  Character Escaping ............................................................................................................................... 805  
POST Upload Example ................................................................................................................................... 807  

---

API Version 2006-03-01
Amazon S3 REST API Introduction

Welcome to the *Amazon Simple Storage Service API Reference*. This guide explains the Amazon Simple Storage Service (Amazon S3) application programming interface (API). It describes various API operations, related request and response structures, and error codes. The current version of the Amazon S3 API is 2006-03-01.

Amazon S3 supports the REST API.

**Note**

Support for SOAP over HTTP is deprecated, but it is still available over HTTPS. However, new Amazon S3 features will not be supported for SOAP. We recommend that you use either the REST API or the AWS SDKs.

Read the following about authentication and access control before going to specific API topics.

Requests to Amazon S3 can be authenticated or anonymous. Authenticated access requires credentials that AWS can use to authenticate your requests. When making REST API calls directly from your code, you create a signature using valid credentials and include the signature in your request. For information about various authentication methods and signature calculations, see *Authenticating Requests (AWS Signature Version 4)* (p. 733).

Making REST API calls directly from your code can be cumbersome. It requires you to write the necessary code to calculate a valid signature to authenticate your requests. We recommend the following alternatives instead:

- Use the AWS SDKs to send your requests (see *Sample Code and Libraries*). With this option, you don't need to write code to calculate a signature for request authentication because the SDK clients authenticate your requests by using access keys that you provide. Unless you have a good reason not to, you should always use the AWS SDKs.
- Use the AWS CLI to make Amazon S3 API calls. For information about setting up the AWS CLI and example Amazon S3 commands see the following topics:

  **Set Up the AWS CLI** in the *Amazon Simple Storage Service Developer Guide*.

  **Using Amazon S3 with the AWS Command Line Interface** in the *AWS Command Line Interface User Guide*.

You can have valid credentials to authenticate your requests, but unless you have permissions you cannot create or access Amazon S3 resources. For example, you must have permissions to create an S3 bucket or get an object from your bucket. If you use root credentials of your AWS account, you have all the permissions. However, using root credentials is not recommended. Instead, we recommend that you create IAM users in your account and manage user permissions. For more information, see *Managing Access Permissions to Your Amazon S3 Resources* in the *Amazon Simple Storage Service Developer Guide*. 
Amazon S3 API Reference

This section contains the Amazon S3 API Reference documentation. The Amazon S3 APIs are grouped into two sets: Amazon Simple Storage Service and AWS S3 Control. There is no functional distinction between the two sets. In general, APIs that apply bucket- and object-level actions are in the Amazon Simple Storage Service set, and APIs that apply account-level actions are in the AWS S3 Control set. If you don't find an API that you're looking for in one set, check the other.

Actions

The following actions are supported by Amazon Simple Storage Service:

- AbortMultipartUpload (p. 8)
- CompleteMultipartUpload (p. 11)
- CopyObject (p. 17)
- CreateBucket (p. 28)
- CreateMultipartUpload (p. 33)
- DeleteBucket (p. 43)
- DeleteBucketAnalyticsConfiguration (p. 45)
- DeleteBucketCors (p. 47)
- DeleteBucketEncryption (p. 49)
- DeleteBucketIntelligentTieringConfiguration (p. 51)
- DeleteBucketInventoryConfiguration (p. 53)
- DeleteBucketLifecycle (p. 55)
- DeleteBucketMetricsConfiguration (p. 57)
- DeleteBucketOwnershipControls (p. 59)
- DeleteBucketPolicy (p. 61)
- DeleteBucketReplication (p. 63)
- DeleteBucketTagging (p. 65)
- DeleteBucketWebsite (p. 67)
- DeleteObject (p. 69)
- DeleteObjects (p. 74)
- DeleteObjectTagging (p. 82)
- DeletePublicAccessBlock (p. 85)
- GetBucketAccelerateConfiguration (p. 87)
- GetBucketAcl (p. 90)
- GetBucketAnalyticsConfiguration (p. 93)
- GetBucketCors (p. 97)
- GetBucketEncryption (p. 100)
- GetBucketIntelligentTieringConfiguration (p. 103)
- GetBucketInventoryConfiguration (p. 106)
- GetBucketLifecycle (p. 110)
- GetBucketLifecycleConfiguration (p. 113)
- GetBucketLocation (p. 117)
• GetBucketLogging (p. 119)
• GetBucketMetricsConfiguration (p. 122)
• GetBucketNotification (p. 126)
• GetBucketNotificationConfiguration (p. 128)
• GetBucketOwnershipControls (p. 131)
• GetBucketPolicy (p. 134)
• GetBucketPolicyStatus (p. 137)
• GetBucketReplication (p. 139)
• GetBucketRequestPayment (p. 143)
• GetBucketTagging (p. 145)
• GetBucketVersioning (p. 148)
• GetBucketWebsite (p. 151)
• GetObject (p. 154)
• GetObjectAcl (p. 166)
• GetObjectLegalHold (p. 170)
• GetObjectLockConfiguration (p. 172)
• GetObjectRetention (p. 174)
• GetObjectTagging (p. 176)
• GetObjectTorrent (p. 179)
• GetPublicAccessBlock (p. 182)
• HeadBucket (p. 185)
• HeadObject (p. 187)
• ListBucketAnalyticsConfigurations (p. 196)
• ListBucketIntelligentTieringConfigurations (p. 200)
• ListBucketInventoryConfigurations (p. 203)
• ListBucketMetricsConfigurations (p. 208)
• ListBuckets (p. 212)
• ListMultipartUploads (p. 214)
• ListObjects (p. 222)
• ListObjectsV2 (p. 230)
• ListObjectVersions (p. 240)
• ListParts (p. 252)
• PutBucketAccelerateConfiguration (p. 258)
• PutBucketAcl (p. 261)
• PutBucketAnalyticsConfiguration (p. 267)
• PutBucketCors (p. 271)
• PutBucketEncryption (p. 275)
• PutBucketIntelligentTieringConfiguration (p. 278)
• PutBucketInventoryConfiguration (p. 281)
• PutBucketLifecycle (p. 286)
• PutBucketLifecycleConfiguration (p. 292)
• PutBucketLogging (p. 298)
• PutBucketMetricsConfiguration (p. 302)
• PutBucketNotification (p. 306)
• PutBucketNotificationConfiguration (p. 308)
• PutBucketOwnershipControls (p. 314)
• PutBucketPolicy (p. 317)
• PutBucketReplication (p. 320)
• PutBucketRequestPayment (p. 326)
• PutBucketTagging (p. 329)
• PutBucketVersioning (p. 332)
• PutBucketWebsite (p. 336)
• PutObject (p. 343)
• PutObjectAcl (p. 354)
• PutObjectLegalHold (p. 361)
• PutObjectLockConfiguration (p. 364)
• PutObjectRetention (p. 367)
• PutObjectTagging (p. 370)
• PutPublicAccessBlock (p. 374)
• RestoreObject (p. 378)
• SelectObjectContent (p. 387)
• UploadPart (p. 395)
• UploadPartCopy (p. 401)

The following actions are supported by AWS S3 Control:

• CreateAccessPoint (p. 409)
• CreateBucket (p. 413)
• CreateJob (p. 417)
• DeleteAccessPoint (p. 422)
• DeleteAccessPointPolicy (p. 424)
• DeleteBucket (p. 426)
• DeleteBucketLifecycleConfiguration (p. 428)
• DeleteBucketPolicy (p. 430)
• DeleteBucketTagging (p. 432)
• DeleteJobTagging (p. 434)
• DeletePublicAccessBlock (p. 436)
• DescribeJob (p. 438)
• GetAccessPoint (p. 442)
• GetAccessPointPolicy (p. 445)
• GetAccessPointPolicyStatus (p. 447)
• GetBucket (p. 449)
• GetBucketLifecycleConfiguration (p. 452)
• GetBucketPolicy (p. 456)
• GetBucketTagging (p. 459)
• GetJobTagging (p. 462)
• GetPublicAccessBlock (p. 464)
• ListAccessPoints (p. 467)
• ListJobs (p. 470)
• ListRegionalBuckets (p. 473)
• PutAccessPointPolicy (p. 476)
• PutBucketLifecycleConfiguration (p. 479)
• PutBucketPolicy (p. 482)
• PutBucketTagging (p. 485)
• PutJobTagging (p. 488)
• PutPublicAccessBlock (p. 491)
• UpdateJobPriority (p. 494)
• UpdateJobStatus (p. 496)

The following actions are supported by Amazon S3 on Outposts:

• CreateEndpoint (p. 499)
• DeleteEndpoint (p. 502)
• ListEndpoints (p. 504)

Amazon Simple Storage Service

The following actions are supported by Amazon Simple Storage Service:

• AbortMultipartUpload (p. 8)
• CompleteMultipartUpload (p. 11)
• CopyObject (p. 17)
• CreateBucket (p. 28)
• CreateMultipartUpload (p. 33)
• DeleteBucket (p. 43)
• DeleteBucketAnalyticsConfiguration (p. 45)
• DeleteBucketCors (p. 47)
• DeleteBucketEncryption (p. 49)
• DeleteBucketIntelligentTieringConfiguration (p. 51)
• DeleteBucketInventoryConfiguration (p. 53)
• DeleteBucketLifecycle (p. 55)
• DeleteBucketMetricsConfiguration (p. 57)
• DeleteBucketOwnershipControls (p. 59)
• DeleteBucketPolicy (p. 61)
• DeleteBucketReplication (p. 63)
• DeleteBucketTagging (p. 65)
• DeleteBucketWebsite (p. 67)
• DeleteObject (p. 69)
• DeleteObjects (p. 74)
• DeleteObjectTagging (p. 82)
• DeletePublicAccessBlock (p. 85)
• GetBucketAccelerateConfiguration (p. 87)
• GetBucketAcl (p. 90)
• GetBucketAnalyticsConfiguration (p. 93)
• GetBucketCors (p. 97)
• GetBucketEncryption (p. 100)
• GetBucketIntelligentTieringConfiguration (p. 103)
• GetBucketInventoryConfiguration (p. 106)
• GetBucketLifecycle (p. 110)
• GetBucketLifecycleConfiguration (p. 113)
• GetBucketLocation (p. 117)
• GetBucketLogging (p. 119)
• GetBucketMetricsConfiguration (p. 122)
• GetBucketNotification (p. 126)
• GetBucketNotificationConfiguration (p. 128)
• GetBucketOwnershipControls (p. 131)
• GetBucketPolicy (p. 134)
• GetBucketPolicyStatus (p. 137)
• GetBucketReplication (p. 139)
• GetBucketRequestPayment (p. 143)
• GetBucketTagging (p. 145)
• GetBucketVersioning (p. 148)
• GetBucketWebsite (p. 151)
• GetObject (p. 154)
• GetObjectAcl (p. 166)
• GetObjectLegalHold (p. 170)
• GetObjectLockConfiguration (p. 172)
• GetObjectRetention (p. 174)
• GetObjectTagging (p. 176)
• GetObjectTorrent (p. 179)
• GetPublicAccessBlock (p. 182)
• HeadBucket (p. 185)
• HeadObject (p. 187)
• ListBucketAnalyticsConfigurations (p. 196)
• ListBucketIntelligentTieringConfigurations (p. 200)
• ListBucketInventoryConfigurations (p. 203)
• ListBucketMetricsConfigurations (p. 208)
• ListBuckets (p. 212)
• ListMultipartUploads (p. 214)
• ListObjects (p. 222)
• ListObjectsV2 (p. 230)
• ListObjectVersions (p. 240)
• ListParts (p. 252)
• PutBucketAccelerateConfiguration (p. 258)
• PutBucketAcl (p. 261)
• PutBucketAnalyticsConfiguration (p. 267)
• PutBucketCors (p. 271)
• PutBucketEncryption (p. 275)
• PutBucketIntelligentTieringConfiguration (p. 278)
• PutBucketInventoryConfiguration (p. 281)
• PutBucketLifecycle (p. 286)
• PutBucketLifecycleConfiguration (p. 292)
• PutBucketLogging (p. 298)
• PutBucketMetricsConfiguration (p. 302)
• PutBucketNotification (p. 306)
• PutBucketNotificationConfiguration (p. 308)
• PutBucketOwnershipControls (p. 314)
• PutBucketPolicy (p. 317)
• PutBucketReplication (p. 320)
• PutBucketRequestPayment (p. 326)
• PutBucketTagging (p. 329)
• PutBucketVersioning (p. 332)
• PutBucketWebsite (p. 336)
• PutObject (p. 343)
• PutObjectAcl (p. 354)
• PutObjectLegalHold (p. 361)
• PutObjectLockConfiguration (p. 364)
• PutObjectRetention (p. 367)
• PutObjectTagging (p. 370)
• PutPublicAccessBlock (p. 374)
• RestoreObject (p. 378)
• SelectObjectContent (p. 387)
• UploadPart (p. 395)
• UploadPartCopy (p. 401)
AbortMultipartUpload
Service: Amazon Simple Storage Service

This operation aborts a multipart upload. After a multipart upload is aborted, no additional parts can be
uploaded using that upload ID. The storage consumed by any previously uploaded parts will be freed.
However, if any part uploads are currently in progress, those part uploads might or might not succeed.
As a result, it might be necessary to abort a given multipart upload multiple times in order to completely
free all storage consumed by all parts.

To verify that all parts have been removed, so you don’t get charged for the part storage, you should call
the ListParts operation and ensure that the parts list is empty.

For information about permissions required to use the multipart upload API, see Multipart Upload API
and Permissions.

The following operations are related to AbortMultipartUpload:

- CreateMultipartUpload
- UploadPart
- CompleteMultipartUpload
- ListParts
- ListMultipartUploads

Request Syntax

DELETE /Key+?uploadId={UploadId} HTTP/1.1
Host: {Bucket}.s3.amazonaws.com
x-amz-request-payer: {RequestPayer}
x-amz-expected-bucket-owner: {ExpectedBucketOwner}

URI Request Parameters

The request uses the following URI parameters.

Bucket (p. 8)

The bucket name to which the upload was taking place.

When using this API with an access point, you must direct requests to the access point
hostname. The access point hostname takes the form AccessPointName-AccountId.s3-
accesspoint.Region.amazonaws.com. When using this operation with an access point through the
AWS SDKs, you provide the access point ARN in place of the bucket name. For more information
about access point ARNs, see Using Access Points in the Amazon Simple Storage Service Developer
Guide.

When using this API with Amazon S3 on Outposts, you must direct requests to the S3 on Outposts
hostname. The S3 on Outposts hostname takes the form AccessPointName-AccountId.outpostId.s3-
outposts.Region.amazonaws.com. When using this operation using S3 on Outposts through the
AWS SDKs, you provide the Outposts bucket ARN in place of the bucket name. For more information
about S3 on Outposts ARNs, see Using S3 on Outposts in the Amazon Simple Storage Service
Developer Guide.

Required: Yes

Key (p. 8)

Key of the object for which the multipart upload was initiated.
Length Constraints: Minimum length of 1.

Required: Yes

uploadID (p. 8)

Upload ID that identifies the multipart upload.

Required: Yes

x-amz-expected-bucket-owner (p. 8)

The account ID of the expected bucket owner. If the bucket is owned by a different account, the request will fail with an HTTP 403 (Access Denied) error.

x-amz-request-payer (p. 8)

Confirms that the requester knows that they will be charged for the request. Bucket owners need not specify this parameter in their requests. For information about downloading objects from requester pays buckets, see Downloading Objects in Requestor Pays Buckets in the Amazon S3 Developer Guide.

Valid Values: requester

Request Body

The request does not have a request body.

Response Syntax

<table>
<thead>
<tr>
<th>HTTP/1.1 204</th>
</tr>
</thead>
<tbody>
<tr>
<td>x-amz-request-charged: RequestCharged</td>
</tr>
</tbody>
</table>

Response Elements

If the action is successful, the service sends back an HTTP 204 response.

The response returns the following HTTP headers.

x-amz-request-charged (p. 9)

If present, indicates that the requester was successfully charged for the request.

Valid Values: requester

The specified multipart upload does not exist.

Examples

Sample Request

The following request aborts a multipart upload identified by its upload ID.

```plaintext
DELETE /example-object?
uploadId=VXBsb2FkIElEIGZvciBlbHZpbmncncyBteS1tb3ZpZS5tMnRzIHVwbG9hZ HTTP/1.1
Host: example-bucket.s3.<Region>.amazonaws.com
Date: Mon, 1 Nov 2010 20:34:56 GMT
Authorization: authorization string
```
Sample Response

This example illustrates one usage of AbortMultipartUpload.

```
HTTP/1.1 204 OK
x-amz-id-2: Weag1LlByRx9e6j5Onimru9p04ZVKnJ2qz7/C1NPcfTWAtRPFfTaOFg==
x-amz-request-id: 996c76696e6727732072657175657374
Date: Mon, 1 Nov 2010 20:34:56 GMT
Content-Length: 0
Connection: keep-alive
Server: AmazonS3
```

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3
CompleteMultipartUpload
Service: Amazon Simple Storage Service

Completes a multipart upload by assembling previously uploaded parts.

You first initiate the multipart upload and then upload all parts using the UploadPart operation. After successfully uploading all relevant parts of an upload, you call this operation to complete the upload. Upon receiving this request, Amazon S3 concatenates all the parts in ascending order by part number to create a new object. In the Complete Multipart Upload request, you must provide the parts list. You must ensure that the parts list is complete. This operation concatenates the parts that you provide in the list. For each part in the list, you must provide the part number and the ETag value, returned after that part was uploaded.

Processing of a Complete Multipart Upload request could take several minutes to complete. After Amazon S3 begins processing the request, it sends an HTTP response header that specifies a 200 OK response. While processing is in progress, Amazon S3 periodically sends white space characters to keep the connection from timing out. Because a request could fail after the initial 200 OK response has been sent, it is important that you check the response body to determine whether the request succeeded.

Note that if CompleteMultipartUpload fails, applications should be prepared to retry the failed requests. For more information, see Amazon S3 Error Best Practices.

For more information about multipart uploads, see Uploading Objects Using Multipart Upload.

For information about permissions required to use the multipart upload API, see Multipart Upload API and Permissions.

CompleteMultipartUpload has the following special errors:

- Error code: EntityTooSmall
  - Description: Your proposed upload is smaller than the minimum allowed object size. Each part must be at least 5 MB in size, except the last part.
  - 400 Bad Request

- Error code: InvalidPart
  - Description: One or more of the specified parts could not be found. The part might not have been uploaded, or the specified entity tag might not have matched the part's entity tag.
  - 400 Bad Request

- Error code: InvalidPartOrder
  - Description: The list of parts was not in ascending order. The parts list must be specified in order by part number.
  - 400 Bad Request

- Error code: NoSuchUpload
  - Description: The specified multipart upload does not exist. The upload ID might be invalid, or the multipart upload might have been aborted or completed.
  - 404 Not Found

The following operations are related to CompleteMultipartUpload:

- CreateMultipartUpload
- UploadPart
- AbortMultipartUpload
- ListParts
- ListMultipartUploads
**Request Syntax**

```
POST /Key?uploadId=UploadId HTTP/1.1
Host: Bucket.s3.amazonaws.com
x-amz-request-payer: RequestPayer
x-amz-expected-bucket-owner: ExpectedBucketOwner
<?xml version="1.0" encoding="UTF-8"?>
  <Part>
    <ETag>string</ETag>
    <PartNumber>integer</PartNumber>
  </Part>
  ...
</CompleteMultipartUpload>
```

**URI Request Parameters**

The request uses the following URI parameters.

**Bucket (p. 12)**

Name of the bucket to which the multipart upload was initiated.

Required: Yes

**Key (p. 12)**

Object key for which the multipart upload was initiated.

Length Constraints: Minimum length of 1.

Required: Yes

**uploadId (p. 12)**

ID for the initiated multipart upload.

Required: Yes

**x-amz-expected-bucket-owner (p. 12)**

The account id of the expected bucket owner. If the bucket is owned by a different account, the request will fail with an HTTP 403 (Access Denied) error.

**x-amz-request-payer (p. 12)**

Confirms that the requester knows that they will be charged for the request. Bucket owners need not specify this parameter in their requests. For information about downloading objects from requester pays buckets, see Downloading Objects in Requestor Pays Buckets in the Amazon S3 Developer Guide.

Valid Values: requester

**Request Body**

The request accepts the following data in XML format.

**CompleteMultipartUpload (p. 12)**

Root level tag for the CompleteMultipartUpload parameters.

Required: Yes
Part (p. 12)

Array of CompletedPart data types.

Type: Array of CompletedPart (p. 530) data types

Required: No

Response Syntax

```
HTTP/1.1 200
x-amz-expiration: Expiration
x-amz-server-side-encryption: ServerSideEncryption
x-amz-version-id: VersionId
x-amz-server-side-encryption-aws-kms-key-id: SSEKMSKeyId
x-amz-request-charged: RequestCharged
<?xml version="1.0" encoding="UTF-8"?>
<CompleteMultipartUploadResult>
  <Location>string</Location>
  <Bucket>string</Bucket>
  <Key>string</Key>
  <ETag>string</ETag>
</CompleteMultipartUploadResult>
```

Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The response returns the following HTTP headers.

**x-amz-expiration (p. 13)**

If the object expiration is configured, this will contain the expiration date (expiry-date) and rule ID (rule-id). The value of rule-id is URL encoded.

**x-amz-request-charged (p. 13)**

If present, indicates that the requester was successfully charged for the request.

Valid Values: requester

**x-amz-server-side-encryption (p. 13)**

If you specified server-side encryption either with an Amazon S3-managed encryption key or an AWS KMS customer master key (CMK) in your initiate multipart upload request, the response includes this header. It confirms the encryption algorithm that Amazon S3 used to encrypt the object.

Valid Values: AES256 | aws:kms

**x-amz-server-side-encryption-aws-kms-key-id (p. 13)**

If present, specifies the ID of the AWS Key Management Service (AWS KMS) symmetric customer managed customer master key (CMK) that was used for the object.

**x-amz-version-id (p. 13)**

Version ID of the newly created object, in case the bucket has versioning turned on.

The following data is returned in XML format by the service.
CompleteMultipartUploadResult (p. 13)

Root level tag for the CompleteMultipartUploadResult parameters.

Required: Yes

Bucket (p. 13)

The name of the bucket that contains the newly created object.

When using this API with an access point, you must direct requests to the access point hostname. The access point hostname takes the form AccessPointName-AccountId.s3-accesspoint.Region.amazonaws.com. When using this operation with an access point through the AWS SDKs, you provide the access point ARN in place of the bucket name. For more information about access point ARNs, see Using Access Points in the Amazon Simple Storage Service Developer Guide.

When using this API with Amazon S3 on Outposts, you must direct requests to the S3 on Outposts hostname. The S3 on Outposts hostname takes the form AccessPointName-AccountId.outpostId.s3-outposts.Region.amazonaws.com. When using this operation using S3 on Outposts through the AWS SDKs, you provide the Outposts bucket ARN in place of the bucket name. For more information about S3 on Outposts ARNs, see Using S3 on Outposts in the Amazon Simple Storage Service Developer Guide.

Type: String

ETag (p. 13)

Entity tag that identifies the newly created object's data. Objects with different object data will have different entity tags. The entity tag is an opaque string. The entity tag may or may not be an MD5 digest of the object data. If the entity tag is not an MD5 digest of the object data, it will contain one or more nonhexadecimal characters and/or will consist of less than 32 or more than 32 hexadecimal digits.

Type: String

Key (p. 13)

The object key of the newly created object.

Type: String

Length Constraints: Minimum length of 1.

Location (p. 13)

The URI that identifies the newly created object.

Type: String

Examples

Sample Request

The following Complete Multipart Upload request specifies three parts in the CompleteMultipartUpload element.

POST /example-object?
uploadId=AAAsb2FkIElEIGZvciBlbHZpbmcncyWeS1tb3ZpZS5tMnRzIRRwbG9hZA HTTP/1.1
Host: example-bucket.s3.<Region>.amazonaws.com
Date: Mon, 1 Nov 2010 20:34:56 GMT
Sample Response

The following response indicates that an object was successfully assembled.

HTTP/1.1 200 OK
x-amz-id-2: Uuag1LuByRx9e6j5Onimru9p04ZVKnJ2Qz7/C1NPcfTWATRPfTaOFg==
x-amz-request-id: 656c76696e6727732072657175657374
Date: Mon, 1 Nov 2010 20:34:56 GMT
Connection: close
Server: AmazonS3

<?xml version="1.0" encoding="UTF-8"?>
  <Location>http://Example-Bucket.s3.<Region>.amazonaws.com/Example-Object</Location>
  <Bucket>Example-Bucket</Bucket>
  <Key>Example-Object</Key>
  <ETag>"3858f62230ac3c915f300c664312c11f-9"</ETag>
</CompleteMultipartUploadResult>

Sample Response: Error specified in header

The following response indicates that an error occurred before the HTTP response header was sent.

HTTP/1.1 403 Forbidden
x-amz-id-2: Uuag1LuByRx9e6j5Onimru9p04ZVKnJ2Qz7/C1NPcfTWATRPfTaOFg==
x-amz-request-id: 656c76696e6727732072657175657374
Date: Mon, 1 Nov 2010 20:34:56 GMT
Content-Length: 237
Connection: keep-alive
Server: AmazonS3

<?xml version="1.0" encoding="UTF-8"?>
<Error>
  <Code>AccessDenied</Code>
  <Message>Access Denied</Message>
  <RequestId>656c76696e6727732072657175657374</RequestId>
  <HostId>Uuag1LuByRx9e6j5Onimru9p04ZVKnJ2Qz7/C1NPcfTWATRPfTaOFg==</HostId>
</Error>
Sample Response: Error specified in body

The following response indicates that an error occurred after the HTTP response header was sent. Note that while the HTTP status code is 200 OK, the request actually failed as described in the Error element.

```
HTTP/1.1 200 OK
x-amz-id-2: Uuag1LuByRx9e6jS0nimru9p04ZVKnJ2Qz7/C1NPcfTWAtRPfTaOFg==
x-amz-request-id: 656c76696e6727732072657175657374
Date: Mon, 1 Nov 2010 20:34:56 GMT
Connection: close
Server: AmazonS3

<?xml version="1.0" encoding="UTF-8"?>

<Error>
  <Code>InternalError</Code>
  <Message>We encountered an internal error. Please try again.</Message>
  <RequestId>656c76696e6727732072657175657374</RequestId>
  <HostId>Uuag1LuByRx9e6jS0nimru9p04ZVKnJ2Qz7/C1NPcfTWAtRPfTaOFg==</HostId>
</Error>
```

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3
CopyObject
Service: Amazon Simple Storage Service

Creates a copy of an object that is already stored in Amazon S3.

Note
You can store individual objects of up to 5 TB in Amazon S3. You create a copy of your object up to 5 GB in size in a single atomic operation using this API. However, to copy an object greater than 5 GB, you must use the multipart upload Upload Part - Copy API. For more information, see Copy Object Using the REST Multipart Upload API.

All copy requests must be authenticated. Additionally, you must have read access to the source object and write access to the destination bucket. For more information, see REST Authentication. Both the Region that you want to copy the object from and the Region that you want to copy the object to must be enabled for your account.

A copy request might return an error when Amazon S3 receives the copy request or while Amazon S3 is copying the files. If the error occurs before the copy operation starts, you receive a standard Amazon S3 error. If the error occurs during the copy operation, the error response is embedded in the 200 OK response. This means that a 200 OK response can contain either a success or an error. Design your application to parse the contents of the response and handle it appropriately.

If the copy is successful, you receive a response with information about the copied object.

Note
If the request is an HTTP 1.1 request, the response is chunk encoded. If it were not, it would not contain the content-length, and you would need to read the entire body.

The copy request charge is based on the storage class and Region that you specify for the destination object. For pricing information, see Amazon S3 pricing.

Important
Amazon S3 transfer acceleration does not support cross-Region copies. If you request a cross-Region copy using a transfer acceleration endpoint, you get a 400 Bad Request error. For more information, see Transfer Acceleration.

Metadata
When copying an object, you can preserve all metadata (default) or specify new metadata. However, the ACL is not preserved and is set to private for the user making the request. To override the default ACL setting, specify a new ACL when generating a copy request. For more information, see Using ACLs.

To specify whether you want the object metadata copied from the source object or replaced with metadata provided in the request, you can optionally add the x-amz-metadata-directive header. When you grant permissions, you can use the s3:x-amz-metadata-directive condition key to enforce certain metadata behavior when objects are uploaded. For more information, see Specifying Conditions in a Policy in the Amazon S3 Developer Guide. For a complete list of Amazon S3-specific condition keys, see Actions, Resources, and Condition Keys for Amazon S3.

x-amz-copy-source-if Headers
To only copy an object under certain conditions, such as whether the Etag matches or whether the object was modified before or after a specified date, use the following request parameters:

• x-amz-copy-source-if-match
• x-amz-copy-source-if-none-match
• x-amz-copy-source-if-unmodified-since
• x-amz-copy-source-if-modified-since
If both the `x-amz-copy-source-if-match` and `x-amz-copy-source-if-unmodified-since` headers are present in the request and evaluate as follows, Amazon S3 returns **200 OK** and copies the data:

- `x-amz-copy-source-if-match` condition evaluates to true
- `x-amz-copy-source-if-unmodified-since` condition evaluates to false

If both the `x-amz-copy-source-if-none-match` and `x-amz-copy-source-if-modified-since` headers are present in the request and evaluate as follows, Amazon S3 returns the **412 Precondition Failed** response code:

- `x-amz-copy-source-if-none-match` condition evaluates to false
- `x-amz-copy-source-if-modified-since` condition evaluates to true

**Note**

All headers with the `x-amz-` prefix, including `x-amz-copy-source`, must be signed.

**Encryption**

The source object that you are copying can be encrypted or unencrypted. The source object can be encrypted with server-side encryption using AWS managed encryption keys (SSE-S3 or SSE-KMS) or by using a customer-provided encryption key. With server-side encryption, Amazon S3 encrypts your data as it writes it to disks in its data centers and decrypts the data when you access it.

You can optionally use the appropriate encryption-related headers to request server-side encryption for the target object. You have the option to provide your own encryption key or use SSE-S3 or SSE-KMS, regardless of the form of server-side encryption that was used to encrypt the source object. You can even request encryption if the source object was not encrypted. For more information about server-side encryption, see Using Server-Side Encryption.

**Access Control List (ACL)-Specific Request Headers**

When copying an object, you can optionally use headers to grant ACL-based permissions. By default, all objects are private. Only the owner has full access control. When adding a new object, you can grant permissions to individual AWS accounts or to predefined groups defined by Amazon S3. These permissions are then added to the ACL on the object. For more information, see Access Control List (ACL) Overview and Managing ACLs Using the REST API.

**Storage Class Options**

You can use the `CopyObject` operation to change the storage class of an object that is already stored in Amazon S3 using the `StorageClass` parameter. For more information, see Storage Classes in the Amazon S3 Service Developer Guide.

**Versioning**

By default, `x-amz-copy-source` identifies the current version of an object to copy. If the current version is a delete marker, Amazon S3 behaves as if the object was deleted. To copy a different version, use the `versionId` subresource.

If you enable versioning on the target bucket, Amazon S3 generates a unique version ID for the object being copied. This version ID is different from the version ID of the source object. Amazon S3 returns the version ID of the copied object in the `x-amz-version-id` response header in the response.

If you do not enable versioning or suspend it on the target bucket, the version ID that Amazon S3 generates is always null.
If the source object's storage class is GLACIER, you must restore a copy of this object before you can use it as a source object for the copy operation. For more information, see RestoreObject.

The following operations are related to CopyObject:

- PutObject
- GetObject

For more information, see Copying Objects.

**Request Syntax**

```
PUT /Key HTTP/1.1
Host: Bucket.s3.amazonaws.com
x-amz-acl: ACL
Cache-Control: ContentDisposition
Content-Disposition: ContentEncoding
Content-Language: ContentLanguage
Content-Type: ContentType
x-amz-copy-source: CopySource
x-amz-copy-source-if-match: CopySourceIfMatch
x-amz-copy-source-if-modified-since: CopySourceIfModifiedSince
x-amz-copy-source-if-none-match: CopySourceIfNoneMatch
x-amz-copy-source-if-unmodified-since: CopySourceIfUnmodifiedSince
Expires: Expires
x-amz-grant-full-control: GrantFullControl
x-amz-grant-read: GrantRead
x-amz-grant-read-acp: GrantReadACP
x-amz-grant-write-acp: GrantWriteACP
x-amz-metadata-directive: MetadataDirective
x-amz-server-side-encryption: ServerSideEncryption
x-amz-storage-class: StorageClass
x-amz-website-redirect-location: WebsiteRedirectLocation
x-amz-server-side-encryption-customer-algorithm: SSECustomerAlgorithm
x-amz-server-side-encryption-customer-key: SSECustomerKey
x-amz-server-side-encryption-customer-key-MD5: SSECustomerKeyMD5
x-amz-server-side-encryption-aws-kms-key-id: SSEKMSKeyId
x-amz-server-side-encryption-context: SSEKMSEncryptionContext
x-amz-copy-source-server-side-encryption-customer-algorithm: CopySourceSSECustomerAlgorithm
x-amz-copy-source-server-side-encryption-customer-key: CopySourceSSECustomerKey
x-amz-copy-source-server-side-encryption-customer-key-MD5: CopySourceSSECustomerKeyMD5
x-amz-request-payer: RequestPayer
x-amz-tagging: Tagging
x-amz-object-lock-mode: ObjectLockMode
x-amz-object-lock-retain-until-date: ObjectLockRetainUntilDate
x-amz-object-lock-legal-hold: ObjectLockLegalHoldStatus
x-amz-expected-bucket-owner: ExpectedBucketOwner
x-amz-source-expected-bucket-owner: ExpectedSourceBucketOwner
```

**URI Request Parameters**

The request uses the following URI parameters.

**Bucket (p. 19)**

The name of the destination bucket.

When using this API with an access point, you must direct requests to the access point hostname. The access point hostname takes the form AccessPointName-AccountId.s3-
When using this operation with an access point through the AWS SDKs, you provide the access point ARN in place of the bucket name. For more information about access point ARNs, see Using Access Points in the Amazon Simple Storage Service Developer Guide.

When using this API with Amazon S3 on Outposts, you must direct requests to the S3 on Outposts hostname. The S3 on Outposts hostname takes the form AccessPointName-AccountId.outpostId.s3-outposts.Region.amazonaws.com. When using this operation using S3 on Outposts through the AWS SDKs, you provide the Outposts bucket ARN in place of the bucket name. For more information about S3 on Outposts ARNs, see Using S3 on Outposts in the Amazon Simple Storage Service Developer Guide.

Required: Yes

**Cache-Control (p. 19)**

Specifies caching behavior along the request/reply chain.

**Content-Disposition (p. 19)**

Specifies presentational information for the object.

**Content-Encoding (p. 19)**

Specifies what content encodings have been applied to the object and thus what decoding mechanisms must be applied to obtain the media-type referenced by the Content-Type header field.

**Content-Language (p. 19)**

The language the content is in.

**Content-Type (p. 19)**

A standard MIME type describing the format of the object data.

**Expires (p. 19)**

The date and time at which the object is no longer cacheable.

**Key (p. 19)**

The key of the destination object.

Length Constraints: Minimum length of 1.

Required: Yes

**x-amz-acl (p. 19)**

The canned ACL to apply to the object.

This action is not supported by Amazon S3 on Outposts.

Valid Values: private | public-read | public-read-write | authenticated-read | aws-exec-read | bucket-owner-read | bucket-owner-full-control

**x-amz-copy-source (p. 19)**

Specifies the source object for the copy operation. You specify the value in one of two formats, depending on whether you want to access the source object through an access point:

- For objects not accessed through an access point, specify the name of the source bucket and the key of the source object, separated by a slash (/). For example, to copy the object reports/january.pdf from the bucket awsexamplebucket, use awsexamplebucket/reports/january.pdf. The value must be URL encoded.
- For objects accessed through access points, specify the Amazon Resource Name (ARN) of the object as accessed through the access point, in the format arn:aws:s3:<Region>:<accountId>:accesspoint/<access-point-name>/object/<key>. For example, to copy...
the object reports/january.pdf through access point my-access-point owned by account 123456789012 in Region us-west-2, use the URL encoding of arn:aws:s3:us-west-2:123456789012:accesspoint/my-access-point/object/reports/january.pdf. The value must be URL encoded.

**Note**
Amazon S3 supports copy operations using access points only when the source and destination buckets are in the same AWS Region.

Alternatively, for objects accessed through Amazon S3 on Outposts, specify the ARN of the object as accessed in the format arn:aws:s3-outposts:<Region>:<account-id>:outpost/<outpost-id>/object/<key>. For example, to copy the object reports/january.pdf through outpost my-outpost owned by account 123456789012 in Region us-west-2, use the URL encoding of arn:aws:s3-outposts:us-west-2:123456789012:outpost/my-outpost/object/reports/january.pdf. The value must be URL encoded.

To copy a specific version of an object, append `?versionId=<version-id>` to the value (for example, awsexamplebucket/reports/january.pdf?versionId=QUpfdndhfd8438MNFDN93jdnJFkdmqnh893). If you don't specify a version ID, Amazon S3 copies the latest version of the source object.

**Pattern:** `\/**.+/**+`

**Required:** Yes

- **x-amz-copy-source-if-match (p. 19)**
  Copies the object if its entity tag (ETag) matches the specified tag.

- **x-amz-copy-source-if-modified-since (p. 19)**
  Copies the object if it has been modified since the specified time.

- **x-amz-copy-source-if-none-match (p. 19)**
  Copies the object if its entity tag (ETag) is different than the specified ETag.

- **x-amz-copy-source-if-unmodified-since (p. 19)**
  Copies the object if it hasn't been modified since the specified time.

- **x-amz-copy-source-server-side-encryption-customer-algorithm (p. 19)**
  Specifies the algorithm to use when decrypting the source object (for example, AES256).

- **x-amz-copy-source-server-side-encryption-customer-key (p. 19)**
  Specifies the customer-provided encryption key for Amazon S3 to use to decrypt the source object. The encryption key provided in this header must be one that was used when the source object was created.

- **x-amz-copy-source-server-side-encryption-customer-key-MD5 (p. 19)**
  Specifies the 128-bit MD5 digest of the encryption key according to RFC 1321. Amazon S3 uses this header for a message integrity check to ensure that the encryption key was transmitted without error.

- **x-amz-expected-bucket-owner (p. 19)**
  The account id of the expected destination bucket owner. If the destination bucket is owned by a different account, the request will fail with an HTTP 403 (Access Denied) error.

- **x-amz-grant-full-control (p. 19)**
  Gives the grantee READ, READ_ACP, and WRITE_ACP permissions on the object.

This action is not supported by Amazon S3 on Outposts.
x-amz-grant-read (p. 19)

Allows grantee to read the object data and its metadata.
This action is not supported by Amazon S3 on Outposts.

x-amz-grant-read-acp (p. 19)

Allows grantee to read the object ACL.
This action is not supported by Amazon S3 on Outposts.

x-amz-grant-write-acp (p. 19)

Allows grantee to write the ACL for the applicable object.
This action is not supported by Amazon S3 on Outposts.

x-amz-metadata-directive (p. 19)

Specifies whether the metadata is copied from the source object or replaced with metadata provided in the request.
Valid Values: COPY | REPLACE

x-amz-object-lock-legal-hold (p. 19)

Specifies whether you want to apply a Legal Hold to the copied object.
Valid Values: ON | OFF

x-amz-object-lock-mode (p. 19)

The Object Lock mode that you want to apply to the copied object.
Valid Values: GOVERNANCE | COMPLIANCE

x-amz-object-lock-retain-until-date (p. 19)

The date and time when you want the copied object's Object Lock to expire.

x-amz-request-payer (p. 19)

Confirms that the requester knows that they will be charged for the request. Bucket owners need not specify this parameter in their requests. For information about downloading objects from requester pays buckets, see Downloading Objects in Requestor Pays Buckets in the Amazon S3 Developer Guide.

Valid Values: requester

x-amz-server-side-encryption (p. 19)

The server-side encryption algorithm used when storing this object in Amazon S3 (for example, AES256, aws:kms).

Valid Values: AES256 | aws:kms

x-amz-server-side-encryption-aws-kms-key-id (p. 19)

Specifies the AWS KMS key ID to use for object encryption. All GET and PUT requests for an object protected by AWS KMS will fail if not made via SSL or using SigV4. For information about configuring using any of the officially supported AWS SDKs and AWS CLI, see Specifying the Signature Version in Request Authentication in the Amazon S3 Developer Guide.

x-amz-server-side-encryption-context (p. 19)

Specifies the AWS KMS Encryption Context to use for object encryption. The value of this header is a base64-encoded UTF-8 string holding JSON with the encryption context key-value pairs.
x-amz-server-side-encryption-customer-algorithm (p. 19)

Specifies the algorithm to use to when encrypting the object (for example, AES256).

x-amz-server-side-encryption-customer-key (p. 19)

Specifies the customer-provided encryption key for Amazon S3 to use in encrypting data. This value is used to store the object and then it is discarded; Amazon S3 does not store the encryption key. The key must be appropriate for use with the algorithm specified in the x-amz-server-side-encryption-customer-algorithm header.

x-amz-server-side-encryption-customer-key-MD5 (p. 19)

Specifies the 128-bit MD5 digest of the encryption key according to RFC 1321. Amazon S3 uses this header for a message integrity check to ensure that the encryption key was transmitted without error.

x-amz-source-expected-bucket-owner (p. 19)

The account id of the expected source bucket owner. If the source bucket is owned by a different account, the request will fail with an HTTP 403 (Access Denied) error.

x-amz-storage-class (p. 19)

By default, Amazon S3 uses the STANDARD Storage Class to store newly created objects. The STANDARD storage class provides high durability and high availability. Depending on performance needs, you can specify a different Storage Class. Amazon S3 on Outposts only uses the OUTPOSTS Storage Class. For more information, see Storage Classes in the Amazon S3 Service Developer Guide.

Valid Values: STANDARD | REDUCED_REDUNDANCY | STANDARD_IA | ONEZONE_IA | INTELLIGENT_TIERING | GLACIER | DEEP_ARCHIVE | OUTPOSTS

x-amz-tagging (p. 19)

The tag-set for the object destination object this value must be used in conjunction with the TaggingDirective. The tag-set must be encoded as URL Query parameters.

x-amz-tagging-directive (p. 19)

Specifies whether the object tag-set are copied from the source object or replaced with tag-set provided in the request.

Valid Values: COPY | REPLACE

x-amz-website-redirect-location (p. 19)

If the bucket is configured as a website, redirects requests for this object to another object in the same bucket or to an external URL. Amazon S3 stores the value of this header in the object metadata.

Request Body

The request does not have a request body.

Response Syntax

```
HTTP/1.1 200
x-amz-expiration: Expiration
x-amz-copy-source-version-id: CopySourceVersionId
x-amz-version-id: VersionId
x-amz-server-side-encryption: ServerSideEncryption
x-amz-server-side-encryption-customer-algorithm: SSECustomerAlgorithm
x-amz-server-side-encryption-customer-key-MD5: SSECustomerKeyMD5
x-amz-server-side-encryption-aws-kms-key-id: SSES키KeyId
```
Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The response returns the following HTTP headers.

**x-amz-copy-source-version-id (p. 23)**

Version of the copied object in the destination bucket.

**x-amz-expiration (p. 23)**

If the object expiration is configured, the response includes this header.

**x-amz-request-charged (p. 23)**

If present, indicates that the requester was successfully charged for the request.

Valid Values: requester

**x-amz-server-side-encryption (p. 23)**

The server-side encryption algorithm used when storing this object in Amazon S3 (for example, AES256, aws:kms).

Valid Values: AES256 | aws:kms

**x-amz-server-side-encryption-aws-kms-key-id (p. 23)**

If present, specifies the ID of the AWS Key Management Service (AWS KMS) symmetric customer managed customer master key (CMK) that was used for the object.

**x-amz-server-side-encryption-context (p. 23)**

If present, specifies the AWS KMS Encryption Context to use for object encryption. The value of this header is a base64-encoded UTF-8 string holding JSON with the encryption context key-value pairs.

**x-amz-server-side-encryption-customer-algorithm (p. 23)**

If server-side encryption with a customer-provided encryption key was requested, the response will include this header confirming the encryption algorithm used.

**x-amz-server-side-encryption-customer-key-MD5 (p. 23)**

If server-side encryption with a customer-provided encryption key was requested, the response will include this header to provide round-trip message integrity verification of the customer-provided encryption key.

**x-amz-version-id (p. 23)**

Version ID of the newly created copy.

The following data is returned in XML format by the service.

**CopyObjectResult (p. 23)**

Root level tag for the CopyObjectResult parameters.
Required: Yes

**ETag (p. 23)**

Returns the ETag of the new object. The ETag reflects only changes to the contents of an object, not its metadata. The source and destination ETag is identical for a successfully copied object.

Type: String

**LastModified (p. 23)**

Returns the date that the object was last modified.

Type: Timestamp

The source object of the COPY operation is not in the active tier and is only stored in Amazon S3 Glacier.

**Examples**

**Sample Request**

This example copies `my-image.jpg` into the `bucket` bucket, with the key name `my-second-image.jpg`.

```
PUT /my-second-image.jpg HTTP/1.1
Host: bucket.s3.<Region>.amazonaws.com
Date: Wed, 28 Oct 2009 22:32:00 GMT
x-amz-copy-source: /bucket/my-image.jpg
Authorization: authorization string
```

**Sample Response**

This example illustrates one usage of CopyObject.

```
HTTP/1.1 200 OK
Connection: close
Date: Wed, 28 Oct 2009 22:32:00 GMT
Server: AmazonS3

<CopyObjectResult>
  <LastModified>2009-10-28T22:32:00</LastModified>
  <ETag>"9b2cf535f27731c974343645a3985328"</ETag>
</CopyObjectResult>
```

**Sample Request: Copying a Specified Version of an Object**

The following request copies the `my-image.jpg` key with the specified version ID, copies it into the `bucket` bucket, and gives it the `my-second-image.jpg` key.

```
PUT /my-second-image.jpg HTTP/1.1
Host: bucket.s3.<Region>.amazonaws.com
```

**API Version 2006-03-01**

25
Success Response: Copying a versioned object into a version-enabled bucket

The following response shows that an object was copied into a target bucket where versioning is enabled.

HTTP/1.1 200 OK
x-amz-id-2: eftixk72aD6Ap51TncgoF8eFidJG9Z/2mkidFfu8y9AS1ed40pIszj7UDNEHGran
x-amz-request-id: 318B6B6C149832E5
x-amz-version-id: QUpfdndhfd6438MNFDN93jdjJFkdmqnh893
x-amz-copy-source-version-id: 09df8234529fjs0dfi0w52935029wefdj
Date: Wed, 28 Oct 2009 22:32:00 GMT
Connection: close
Server: AmazonS3

<?xml version="1.0" encoding="UTF-8"?>
<CopyObjectResult>
  <LastModified>2009-10-28T22:32:00</LastModified>
  <ETag>"9b2cf535f27731c974343645a3985328"</ETag>
</CopyObjectResult>

Success Response: Copying a versioned object into a version-suspended bucket

The following response shows that an object was copied into a target bucket where versioning is suspended. The parameter VersionId does not appear.

HTTP/1.1 200 OK
x-amz-id-2: eftixk72aD6Ap51TncgoF8eFidJG9Z/2mkidFfu8y9AS1ed40pIszj7UDNEHGran
x-amz-request-id: 318B6B6C149832E5
x-amz-copy-source-version-id: 3/L4kqtJlcpXroDTDmJ+rmSpXd3d1brHY+MTRCxf3vjVBH40N8XgdRQBPUMLuO
Date: Wed, 28 Oct 2009 22:32:00 GMT
Connection: close
Server: AmazonS3

<?xml version="1.0" encoding="UTF-8"?>
<CopyObjectResult>
  <LastModified>2009-10-28T22:32:00</LastModified>
  <ETag>"9b2cf535f27731c974343645a3985328"</ETag>
</CopyObjectResult>

Example: Copy from unencrypted object to an object encrypted with server-side encryption with customer-provided encryption keys

The following example specifies the HTTP PUT header to copy an unencrypted object to an object encrypted with server-side encryption with customer-provided encryption keys (SSE-C).

PUT /exampleDestinationObject HTTP/1.1
Host: example-destination-bucket.s3.<Region>.amazonaws.com
x-amz-server-side-encryption-customer-algorithm: AES256
Example: Copy from an object encrypted with SSE-C to an object encrypted with SSE-C

The following example specifies the HTTP PUT header to copy an object encrypted with server-side encryption with customer-provided encryption keys to an object encrypted with server-side encryption with customer-provided encryption keys for key rotation.

```
PUT /exampleDestinationObject HTTP/1.1
Host: example-destination-bucket.s3.<Region>.amazonaws.com
x-amz-server-side-encryption-customer-key: Base64(NewKey)
x-amz-server-side-encryption-customer-key-MD5: Base64(MD5(NewKey))
x-amz-metadata-directive: metadata_directive
x-amz-copy-source: /source_bucket/sourceObject
x-amz-copy-source-if-match: etag
x-amz-copy-source-if-none-match: etag
x-amz-copy-source-if-unmodified-since: time_stamp
x-amz-copy-source-if-modified-since: time_stamp
Base64(MD5(OldKey))
```

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3
CreateBucket
Service: Amazon Simple Storage Service

Creates a new S3 bucket. To create a bucket, you must register with Amazon S3 and have a valid AWS Access Key ID to authenticate requests. Anonymous requests are never allowed to create buckets. By creating the bucket, you become the bucket owner.

Not every string is an acceptable bucket name. For information about bucket naming restrictions, see Working with Amazon S3 buckets.

If you want to create an Amazon S3 on Outposts bucket, see Create Bucket.

By default, the bucket is created in the US East (N. Virginia) Region. You can optionally specify a Region in the request body. You might choose a Region to optimize latency, minimize costs, or address regulatory requirements. For example, if you reside in Europe, you will probably find it advantageous to create buckets in the Europe (Ireland) Region. For more information, see Accessing a bucket.

Note
If you send your create bucket request to the s3.amazonaws.com endpoint, the request goes to the us-east-1 Region. Accordingly, the signature calculations in Signature Version 4 must use us-east-1 as the Region, even if the location constraint in the request specifies another Region where the bucket is to be created. If you create a bucket in a Region other than US East (N. Virginia), your application must be able to handle 307 redirect. For more information, see Virtual hosting of buckets.

When creating a bucket using this operation, you can optionally specify the accounts or groups that should be granted specific permissions on the bucket. There are two ways to grant the appropriate permissions using the request headers.

- Specify a canned ACL using the x-amz-acl request header. Amazon S3 supports a set of predefined ACLs, known as canned ACLs. Each canned ACL has a predefined set of grantees and permissions. For more information, see Canned ACL.
- Specify access permissions explicitly using the x-amz-grant-read, x-amz-grant-write, x-amz-grant-read-acp, x-amz-grant-write-acp, and x-amz-grant-full-control headers. These headers map to the set of permissions Amazon S3 supports in an ACL. For more information, see Access control list (ACL) overview.

You specify each grantee as a type=value pair, where the type is one of the following:
- id – if the value specified is the canonical user ID of an AWS account
- uri – if you are granting permissions to a predefined group
- emailAddress – if the value specified is the email address of an AWS account

Note
Using email addresses to specify a grantee is only supported in the following AWS Regions:
- US East (N. Virginia)
- US West (N. California)
- US West (Oregon)
- Asia Pacific (Singapore)
- Asia Pacific (Sydney)
- Asia Pacific (Tokyo)
- Europe (Ireland)
- South America (São Paulo)

For a list of all the Amazon S3 supported Regions and endpoints, see Regions and Endpoints in the AWS General Reference.

For example, the following x-amz-grant-read header grants the AWS accounts identified by account IDs permissions to read object data and its metadata:
x-amz-grant-read: id="11112222333", id="444455556666"

Note
You can use either a canned ACL or specify access permissions explicitly. You cannot do both.

The following operations are related to CreateBucket:

- PutObject
- DeleteBucket

Request Syntax

```
PUT / HTTP/1.1
Host: Bucket.s3.amazonaws.com
x-amz-acl: ACL
x-amz-grant-full-control: GrantFullControl
x-amz-grant-read: GrantRead
x-amz-grant-read-acp: GrantReadACP
x-amz-grant-write: GrantWrite
x-amz-grant-write-acp: GrantWriteACP
x-amz-bucket-object-lock-enabled: ObjectLockEnabledForBucket
<?xml version="1.0" encoding="UTF-8"?>
<CreateBucketConfiguration xmlns="http://s3.amazonaws.com/doc/2006-03-01/">
  <LocationConstraint>string</LocationConstraint>
</CreateBucketConfiguration>
```

URI Request Parameters

The request uses the following URI parameters.

**Bucket (p. 29)**

The name of the bucket to create.

Required: Yes

**x-amz-acl (p. 29)**

The canned ACL to apply to the bucket.

Valid Values: private | public-read | public-read-write | authenticated-read

**x-amz-bucket-object-lock-enabled (p. 29)**

Specifies whether you want S3 Object Lock to be enabled for the new bucket.

**x-amz-grant-full-control (p. 29)**

Allows grantee the read, write, read ACP, and write ACP permissions on the bucket.

**x-amz-grant-read (p. 29)**

Allows grantee to list the objects in the bucket.

**x-amz-grant-read-acp (p. 29)**

Allows grantee to read the bucket ACL.

**x-amz-grant-write (p. 29)**

Allows grantee to create, overwrite, and delete any object in the bucket.
x-amz-grant-write-acp (p. 29)

Allows grantee to write the ACL for the applicable bucket.

Request Body

The request accepts the following data in XML format.

CreateBucketConfiguration (p. 29)

Root level tag for the CreateBucketConfiguration parameters.

Required: Yes

LocationConstraint (p. 29)

Specifies the Region where the bucket will be created. If you don't specify a Region, the bucket is created in the US East (N. Virginia) Region (us-east-1).

Type: String

Valid Values: af-south-1 | ap-east-1 | ap-northeast-1 | ap-northeast-2 | ap-northeast-3 | ap-south-1 | ap-southeast-1 | ap-southeast-2 | ca-central-1 | cn-north-1 | cn-northwest-1 | EU | eu-central-1 | eu-north-1 | eu-south-1 | eu-west-1 | eu-west-2 | eu-west-3 | me-south-1 | sa-east-1 | us-east-2 | us-gov-east-1 | us-gov-west-1 | us-west-1 | us-west-2

Required: No

Response Syntax

```
HTTP/1.1 200
Location: Location
```

Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The response returns the following HTTP headers.

Location (p. 30)

Specifies the Region where the bucket will be created. If you are creating a bucket on the US East (N. Virginia) Region (us-east-1), you do not need to specify the location.

The requested bucket name is not available. The bucket namespace is shared by all users of the system. Select a different name and try again. The bucket you tried to create already exists, and you own it. Amazon S3 returns this error in all AWS Regions except in the North Virginia Region. For legacy compatibility, if you re-create an existing bucket that you already own in the North Virginia Region, Amazon S3 returns 200 OK and resets the bucket access control lists (ACLs).

Examples

Sample Request

This request creates a bucket named colorpictures.
Sample Response

This example illustrates one usage of CreateBucket.

HTTP/1.1 200 OK
x-amz-id-2: YgIPIfBiKa2bj0KMg95r/0zo3emzU4dZsD4rcKCHQUAdQk3ShJT00pXUueF6QRo
x-amz-request-id: 236A89052468E5A01
Date: Wed, 01 Mar 2006 12:00:00 GMT

Location: /colorpictures
Content-Length: 0
Connection: close
Server: AmazonS3

Sample Request: Setting the Region of a bucket

The following request sets the Region for the bucket to Europe.

PUT / HTTP/1.1
Host: bucketName.s3.amazonaws.com
Date: Wed, 12 Oct 2009 17:50:00 GMT
Authorization: authorization string
Content-Type: text/plain
Content-Length: 124

<CreateBucketConfiguration xmlns="http://s3.amazonaws.com/doc/2006-03-01/">
  <LocationConstraint>Europe</LocationConstraint>
</CreateBucketConfiguration>

Sample Request: Creating a bucket and configuring access permission using a canned ACL

This request creates a bucket named colorpictures and sets the ACL to private.

PUT / HTTP/1.1
Host: colorpictures.s3.<Region>.amazonaws.com
Content-Length: 0
x-amz-acl: private
Date: Wed, 01 Mar 2006 12:00:00 GMT
Authorization: authorization string

Sample Response

This example illustrates one usage of CreateBucket.
Sample Request: Creating a bucket and configuring access permissions explicitly

This request creates a bucket named `colorpictures` and grants WRITE permission to the AWS account identified by an email address.

```
PUT HTTP/1.1
Host: colorpictures.s3.<Region>.amazonaws.com
x-amz-date: Sat, 07 Apr 2012 00:54:40 GMT
Authorization: authorization string
x-amz-grant-write: emailAddress="xyz@amazon.com", emailAddress="abc@amazon.com"
```

Sample Response

This example illustrates one usage of CreateBucket.

```
HTTP/1.1 200 OK
```

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3
CreateMultipartUpload
Service: Amazon Simple Storage Service

This operation initiates a multipart upload and returns an upload ID. This upload ID is used to associate all of the parts in the specific multipart upload. You specify this upload ID in each of your subsequent upload part requests (see UploadPart). You also include this upload ID in the final request to either complete or abort the multipart upload request.

For more information about multipart uploads, see Multipart Upload Overview.

If you have configured a lifecycle rule to abort incomplete multipart uploads, the upload must complete within the number of days specified in the bucket lifecycle configuration. Otherwise, the incomplete multipart upload becomes eligible for an abort operation and Amazon S3 aborts the multipart upload. For more information, see Aborting Incomplete Multipart Uploads Using a Bucket Lifecycle Policy.

For information about the permissions required to use the multipart upload API, see Multipart Upload API and Permissions.

For request signing, multipart upload is just a series of regular requests. You initiate a multipart upload, send one or more requests to upload parts, and then complete the multipart upload process. You sign each request individually. There is nothing special about signing multipart upload requests. For more information about signing, see Authenticating Requests (AWS Signature Version 4).

**Note**
After you initiate a multipart upload and upload one or more parts, to stop being charged for storing the uploaded parts, you must either complete or abort the multipart upload. Amazon S3 frees up the space used to store the parts and stop charging you for storing them only after you either complete or abort a multipart upload.

You can optionally request server-side encryption. For server-side encryption, Amazon S3 encrypts your data as it writes it to disks in its data centers and decrypts it when you access it. You can provide your own encryption key, or use AWS Key Management Service (AWS KMS) customer master keys (CMKs) or Amazon S3-managed encryption keys. If you choose to provide your own encryption key, the request headers you provide in UploadPart and UploadPartCopy requests must match the headers you used in the request to initiate the upload by using CreateMultipartUpload.

To perform a multipart upload with encryption using an AWS KMS CMK, the requester must have permission to the kms:Encrypt, kms:Decrypt, kms:ReEncrypt*, kms:GenerateDataKey*, and kms:DescribeKey actions on the key. These permissions are required because Amazon S3 must decrypt and read data from the encrypted file parts before it completes the multipart upload.

If your AWS Identity and Access Management (IAM) user or role is in the same AWS account as the AWS KMS CMK, then you must have these permissions on the key policy. If your IAM user or role belongs to a different account than the key, then you must have the permissions on both the key policy and your IAM user or role.

For more information, see Protecting Data Using Server-Side Encryption.

**Access Permissions**

When copying an object, you can optionally specify the accounts or groups that should be granted specific permissions on the new object. There are two ways to grant the permissions using the request headers:

- Specify a canned ACL with the x-amz-acl request header. For more information, see Canned ACL.
- Specify access permissions explicitly with the x-amz-grant-read, x-amz-grant-write-acp, and x-amz-grant-full-control headers. These parameters map to the set of permissions that Amazon S3 supports in an ACL. For more information, see Access Control List (ACL) Overview.
You can use either a canned ACL or specify access permissions explicitly. You cannot do both.

Server-Side-Encryption-Specific Request Headers

You can optionally tell Amazon S3 to encrypt data at rest using server-side encryption. Server-side encryption is for data encryption at rest. Amazon S3 encrypts your data as it writes it to disks in its data centers and decrypts it when you access it. The option you use depends on whether you want to use AWS managed encryption keys or provide your own encryption key.

- Use encryption keys managed by Amazon S3 or customer master keys (CMKs) stored in AWS Key Management Service (AWS KMS) – If you want AWS to manage the keys used to encrypt data, specify the following headers in the request.
  - x-amz-server-side-encryption
  - x-amz-server-side-encryption-aws-kms-key-id
  - x-amz-server-side-encryption-context

  **Note**
  If you specify `x-amz-server-side-encryption:aws:kms`, but don't provide `x-amz-server-side-encryption-aws-kms-key-id`, Amazon S3 uses the AWS managed CMK in AWS KMS to protect the data.

  **Important**
  All GET and PUT requests for an object protected by AWS KMS fail if you don't make them with SSL or by using SigV4.

For more information about server-side encryption with CMKs stored in AWS KMS (SSE-KMS), see [Protecting Data Using Server-Side Encryption with CMKs stored in AWS KMS](#).

- Use customer-provided encryption keys – If you want to manage your own encryption keys, provide all the following headers in the request.
  - x-amz-server-side-encryption-customer-algorithm
  - x-amz-server-side-encryption-customer-key
  - x-amz-server-side-encryption-customer-key-MD5

For more information about server-side encryption with CMKs stored in AWS KMS (SSE-KMS), see [Protecting Data Using Server-Side Encryption with CMKs stored in AWS KMS](#).

Access-Control-List (ACL)-Specific Request Headers

You also can use the following access control-related headers with this operation. By default, all objects are private. Only the owner has full access control. When adding a new object, you can grant permissions to individual AWS accounts or to predefined groups defined by Amazon S3. These permissions are then added to the access control list (ACL) on the object. For more information, see [Using ACLs](#). With this operation, you can grant access permissions using one of the following two methods:

- Specify a canned ACL (x-amz-acl) — Amazon S3 supports a set of predefined ACLs, known as canned ACLs. Each canned ACL has a predefined set of grantees and permissions. For more information, see [Canned ACL](#).

- Specify access permissions explicitly — To explicitly grant access permissions to specific AWS accounts or groups, use the following headers. Each header maps to specific permissions that Amazon S3 supports in an ACL. For more information, see [Access Control List (ACL) Overview](#). In the header, you specify a list of grantees who get the specific permission. To grant permissions explicitly, use:
  - x-amz-grant-read
  - x-amz-grant-write
  - x-amz-grant-read-acp
  - x-amz-grant-write-acp
  - x-amz-grant-full-control
You specify each grantee as a type=value pair, where the type is one of the following:

- **id** – if the value specified is the canonical user ID of an AWS account
- **uri** – if you are granting permissions to a predefined group
- **emailAddress** – if the value specified is the email address of an AWS account

**Note**

Using email addresses to specify a grantee is only supported in the following AWS Regions:

- US East (N. Virginia)
- US West (N. California)
- US West (Oregon)
- Asia Pacific (Singapore)
- Asia Pacific (Sydney)
- Asia Pacific (Tokyo)
- Europe (Ireland)
- South America (São Paulo)

For a list of all the Amazon S3 supported Regions and endpoints, see [Regions and Endpoints](https://aws.amazon.com/s3/) in the AWS General Reference.

For example, the following `x-amz-grant-read` header grants the AWS accounts identified by account IDs permissions to read object data and its metadata:

```
x-amz-grant-read: id="11112222333", id="444455556666"
```

The following operations are related to `CreateMultipartUpload`:

- **UploadPart**
- **CompleteMultipartUpload**
- **AbortMultipartUpload**
- **ListParts**
- **ListMultipartUploads**

**Request Syntax**

```
POST /{Key+}?uploads HTTP/1.1
Host: Bucket.s3.amazonaws.com
x-amz-acl: ACL
Cache-Control: CacheControl
Content-Disposition: ContentDisposition
Content-Encoding: ContentEncoding
Content-Language: ContentLanguage
Content-Type: ContentType
Expires: Expires
x-amz-grant-full-control: GrantFullControl
x-amz-grant-read: GrantRead
x-amz-grant-read-acp: GrantReadACP
x-amz-grant-write-acp: GrantWriteACP
x-amz-server-side-encryption: ServerSideEncryption
x-amz-storage-class: StorageClass
x-amz-website-redirect-location: WebsiteRedirectLocation
x-amz-server-side-encryption-customer-algorithm: SSECustomerAlgorithm
x-amz-server-side-encryption-customer-key: SSECustomerKey
x-amz-server-side-encryption-customer-key-MD5: SSECustomerKeyMD5
x-amz-server-side-encryption-aws-kms-key-id: SSEKMSKeyId
```

API Version 2006-03-01
x-amz-server-side-encryption-context: SSEKMSEncryptionContext
x-amz-request-payer: RequestPayer
x-amz-tagging: Tagging
x-amz-object-lock-mode: ObjectLockMode
x-amz-object-lock-retain-until-date: ObjectLockRetainUntilDate
x-amz-object-lock-legal-hold: ObjectLockLegalHoldStatus
x-amz-expected-bucket-owner: ExpectedBucketOwner

URI Request Parameters

The request uses the following URI parameters.

Bucket (p. 35)

The name of the bucket to which to initiate the upload

When using this API with an access point, you must direct requests to the access point hostname. The access point hostname takes the form AccessPointName-AccountId.s3-accesspoint.Region.amazonaws.com. When using this operation with an access point through the AWS SDKs, you provide the access point ARN in place of the bucket name. For more information about access point ARNs, see Using Access Points in the Amazon Simple Storage Service Developer Guide.

When using this API with Amazon S3 on Outposts, you must direct requests to the S3 on Outposts hostname. The S3 on Outposts hostname takes the form AccessPointName-AccountId.outpostId.s3-outposts.Region.amazonaws.com. When using this operation using S3 on Outposts through the AWS SDKs, you provide the Outposts bucket ARN in place of the bucket name. For more information about S3 on Outposts ARNs, see Using S3 on Outposts in the Amazon Simple Storage Service Developer Guide.

Required: Yes

Cache-Control (p. 35)

Specifies caching behavior along the request/reply chain.

Content-Disposition (p. 35)

Specifies presentational information for the object.

Content-Encoding (p. 35)

Specifies what content encodings have been applied to the object and thus what decoding mechanisms must be applied to obtain the media-type referenced by the Content-Type header field.

Content-Language (p. 35)

The language the content is in.

Content-Type (p. 35)

A standard MIME type describing the format of the object data.

Expires (p. 35)

The date and time at which the object is no longer cacheable.

Key (p. 35)

Object key for which the multipart upload is to be initiated.

Length Constraints: Minimum length of 1.

Required: Yes
x-amz-acl (p. 35)
The canned ACL to apply to the object.
This action is not supported by Amazon S3 on Outposts.
Valid Values: private | public-read | public-read-write | authenticated-read | aws-exec-read | bucket-owner-read | bucket-owner-full-control

x-amz-expected-bucket-owner (p. 35)
The account id of the expected bucket owner. If the bucket is owned by a different account, the request will fail with an HTTP 403 (Access Denied) error.

x-amz-grant-full-control (p. 35)
Gives the grantee READ, READ_ACP, and WRITE_ACP permissions on the object.
This action is not supported by Amazon S3 on Outposts.

x-amz-grant-read (p. 35)
Allows grantee to read the object data and its metadata.
This action is not supported by Amazon S3 on Outposts.

x-amz-grant-read-acp (p. 35)
Allows grantee to read the object ACL.
This action is not supported by Amazon S3 on Outposts.

x-amz-grant-write-acp (p. 35)
Allows grantee to write the ACL for the applicable object.
This action is not supported by Amazon S3 on Outposts.

x-amz-object-lock-legal-hold (p. 35)
Specifies whether you want to apply a Legal Hold to the uploaded object.
Valid Values: ON | OFF

x-amz-object-lock-mode (p. 35)
Specifies the Object Lock mode that you want to apply to the uploaded object.
Valid Values: GOVERNANCE | COMPLIANCE

x-amz-object-lock-retain-until-date (p. 35)
Specifies the date and time when you want the Object Lock to expire.

x-amz-request-payer (p. 35)
Confirms that the requester knows that they will be charged for the request. Bucket owners need not specify this parameter in their requests. For information about downloading objects from requester pays buckets, see Downloading Objects in Requestor Pays Buckets in the Amazon S3 Developer Guide.
Valid Values: requester

x-amz-server-side-encryption (p. 35)
The server-side encryption algorithm used when storing this object in Amazon S3 (for example, AES256, aws:kms).
Valid Values: AES256 | aws:kms
x-amz-server-side-encryption-aws-kms-key-id (p. 35)

Specifies the ID of the symmetric customer managed AWS KMS CMK to use for object encryption. All GET and PUT requests for an object protected by AWS KMS will fail if not made via SSL or using SigV4. For information about configuring using any of the officially supported AWS SDKs and AWS CLI, see Specifying the Signature Version in Request Authentication in the Amazon S3 Developer Guide.

x-amz-server-side-encryption-context (p. 35)

Specifies the AWS KMS Encryption Context to use for object encryption. The value of this header is a base64-encoded UTF-8 string holding JSON with the encryption context key-value pairs.

x-amz-server-side-encryption-customer-algorithm (p. 35)

Specifies the algorithm to use to when encrypting the object (for example, AES256).

x-amz-server-side-encryption-customer-key (p. 35)

Specifies the customer-provided encryption key for Amazon S3 to use in encrypting data. This value is used to store the object and then it is discarded; Amazon S3 does not store the encryption key. The key must be appropriate for use with the algorithm specified in the x-amz-server-side-encryption-customer-algorithm header.

x-amz-server-side-encryption-customer-key-MD5 (p. 35)

Specifies the 128-bit MD5 digest of the encryption key according to RFC 1321. Amazon S3 uses this header for a message integrity check to ensure that the encryption key was transmitted without error.

x-amz-storage-class (p. 35)

By default, Amazon S3 uses the STANDARD Storage Class to store newly created objects. The STANDARD storage class provides high durability and high availability. Depending on performance needs, you can specify a different Storage Class. Amazon S3 on Outposts only uses the OUTPOSTS Storage Class. For more information, see Storage Classes in the Amazon S3 Service Developer Guide.

Valid Values: STANDARD | REDUCED_REDUNDANCY | STANDARD_IA | ONEZONE_IA | INTELLIGENT_TIERING | GLACIER | DEEP_ARCHIVE | OUTPOSTS

x-amz-tagging (p. 35)

The tag-set for the object. The tag-set must be encoded as URL Query parameters.

x-amz-website-redirect-location (p. 35)

If the bucket is configured as a website, redirects requests for this object to another object in the same bucket or to an external URL. Amazon S3 stores the value of this header in the object metadata.

Request Body

The request does not have a request body.

Response Syntax

HTTP/1.1 200
x-amz-abort-date: AbortDate
x-amz-abort-rule-id: AbortRuleId
x-amz-server-side-encryption: ServerSideEncryption
x-amz-server-side-encryption-customer-algorithm: SSECustomerAlgorithm
x-amz-server-side-encryption-customer-key-md5: SSECustomerKeyMD5
x-amz-server-side-encryption-aws-kms-key-id: SSEKMSKeyId
x-amz-server-side-encryption-context: SSEKMSEncryptionContext
Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The response returns the following HTTP headers.

**x-amz-abort-date (p. 38)**

If the bucket has a lifecycle rule configured with an action to abort incomplete multipart uploads and the prefix in the lifecycle rule matches the object name in the request, the response includes this header. The header indicates when the initiated multipart upload becomes eligible for an abort operation. For more information, see [Aborting Incomplete Multipart Uploads Using a Bucket Lifecycle Policy](#).

The response also includes the `x-amz-abort-rule-id` header that provides the ID of the lifecycle configuration rule that defines this action.

**x-amz-abort-rule-id (p. 38)**

This header is returned along with the `x-amz-abort-date` header. It identifies the applicable lifecycle configuration rule that defines the action to abort incomplete multipart uploads.

**x-amz-request-charged (p. 38)**

If present, indicates that the requester was successfully charged for the request.

Valid Values: requester

**x-amz-server-side-encryption (p. 38)**

The server-side encryption algorithm used when storing this object in Amazon S3 (for example, AES256, aws:kms).

Valid Values: AES256 | aws:kms

**x-amz-server-side-encryption-aws-kms-key-id (p. 38)**

If present, specifies the ID of the AWS Key Management Service (AWS KMS) symmetric customer managed customer master key (CMK) that was used for the object.

**x-amz-server-side-encryption-context (p. 38)**

If present, specifies the AWS KMS Encryption Context to use for object encryption. The value of this header is a base64-encoded UTF-8 string holding JSON with the encryption context key-value pairs.

**x-amz-server-side-encryption-customer-algorithm (p. 38)**

If server-side encryption with a customer-provided encryption key was requested, the response will include this header confirming the encryption algorithm used.

**x-amz-server-side-encryption-customer-key-MD5 (p. 38)**

If server-side encryption with a customer-provided encryption key was requested, the response will include this header to provide round-trip message integrity verification of the customer-provided encryption key.

The following data is returned in XML format by the service.

```xml
<?xml version="1.0" encoding="UTF-8"?>
<InitiateMultipartUploadResult>
  <Bucket>string</Bucket>
  <Key>string</Key>
  <UploadId>string</UploadId>
</InitiateMultipartUploadResult>
```
InitiateMultipartUploadResult (p. 38)

Root level tag for the InitiateMultipartUploadResult parameters.

Required: Yes

Bucket (p. 38)

The name of the bucket to which the multipart upload was initiated.

When using this API with an access point, you must direct requests to the access point hostname. The access point hostname takes the form AccessPointName-AccountId.s3-accesspoint.Region.amazonaws.com. When using this operation with an access point through the AWS SDKs, you provide the access point ARN in place of the bucket name. For more information about access point ARNs, see Using Access Points in the Amazon Simple Storage Service Developer Guide.

When using this API with Amazon S3 on Outposts, you must direct requests to the S3 on Outposts hostname. The S3 on Outposts hostname takes the form AccessPointName-AccountId.outpostId.s3-outposts.Region.amazonaws.com. When using this operation using S3 on Outposts through the AWS SDKs, you provide the Outposts bucket ARN in place of the bucket name. For more information about S3 on Outposts ARNs, see Using S3 on Outposts in the Amazon Simple Storage Service Developer Guide.

Type: String

Key (p. 38)

Object key for which the multipart upload was initiated.

Type: String

Length Constraints: Minimum length of 1.

UploadId (p. 38)

ID for the initiated multipart upload.

Type: String

Examples

Sample Request

This operation initiates a multipart upload for the example-object object.

```
POST /example-object/uploads HTTP/1.1
Host: example-bucket.s3.<Region>.amazonaws.com
Date: Mon, 1 Nov 2010 20:34:56 GMT
Authorization: authorization string
```

Sample Response

This example illustrates one usage of CreateMultipartUpload.

```
HTTP/1.1 200 OK
x-amz-id-2: Uuag1LuByRx9e6j5Onimru9p04ZVKnJ2Qz7/C1NPcfTWAtrRPfTaOFg==
x-amz-request-id: 656c76696e6727732072657175657374
```
Example: Initiate a multipart upload using server-side encryption with customer-provided encryption keys

This example, which initiates a multipart upload request, specifies server-side encryption with customer-provided encryption keys by adding relevant headers.

```
POST /example-object?uploads HTTP/1.1
Host: example-bucket.s3.<Region>.amazonaws.com
Authorization: authorization string
Date: Wed, 28 May 2014 19:34:57 +0000
x-amz-server-side-encryption-customer-key: g0lCfA3Dv40jzSQQj1ukLRgQq7S5d15WEQwC/8SEEX5EXAMPLE
x-amz-server-side-encryption-customer-key-MD5: ZjQrne1X/iTcskbY2example
x-amz-server-side-encryption-customer-algorithm: AES256
```

Sample Response

In the response, Amazon S3 returns an UploadId. In addition, Amazon S3 returns the encryption algorithm and the MD5 digest of the encryption key that you provided in the request.

```
HTTP/1.1 200 OK
x-amz-id-2: 36HRCaIgp57F1FvVvRRVzd3hNn9WoBGfEaCVHTCt8QWf00qxdHasQUGfoXAbhFWD
x-amz-request-id: 50FA1D691B62CA43
x-amz-server-side-encryption-customer-algorithm: AES256
x-amz-server-side-encryption-customer-key-MD5: ZjQrne1X/iTcskbY2m3tFg==
Transfer-Encoding: chunked

<?xml version="1.0" encoding="UTF-8"?><InitiateMultipartUploadResult xmlns="http://s3.amazonaws.com/doc/2006-03-01/">
  <Bucket>example-bucket</Bucket>
  <Key>example-object</Key>
  <UploadId>VXBsb2FkIElEIGZvciA2aWWpbmcncyBteS1b32pZSStMnRzIHVwbG9hZA</UploadId>
</InitiateMultipartUploadResult>
```

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
• AWS SDK for .NET
• AWS SDK for C++
• AWS SDK for Go
• AWS SDK for Java
• AWS SDK for JavaScript
• AWS SDK for PHP V3
• AWS SDK for Python
• AWS SDK for Ruby V3
DeleteBucket
Service: Amazon Simple Storage Service

Deletes the S3 bucket. All objects (including all object versions and delete markers) in the bucket must be deleted before the bucket itself can be deleted.

Related Resources
• CreateBucket
• DeleteObject

Request Syntax

DELETE / HTTP/1.1
Host: Bucket.s3.amazonaws.com
x-amz-expected-bucket-owner: ExpectedBucketOwner

URI Request Parameters

The request uses the following URI parameters.

Bucket (p. 43)
  Specifies the bucket being deleted.
  Required: Yes

x-amz-expected-bucket-owner (p. 43)
  The account id of the expected bucket owner. If the bucket is owned by a different account, the request will fail with an HTTP 403 (Access Denied) error.

Request Body

The request does not have a request body.

Response Syntax

HTTP/1.1 204

Response Elements

If the action is successful, the service sends back an HTTP 204 response with an empty HTTP body.

Examples

Sample Request

This request deletes the bucket named quotes.

DELETE / HTTP/1.1
Host: quotes.s3.<Region>.amazonaws.com
Date: Wed, 01 Mar 2006 12:00:00 GMT
Authorization: authorization string
Sample Response

HTTP/1.1 204 No Content
x-amz-id-2: JuKZqmXuiwFeDQxhD7M8KtsKobSzWA1QEjLbTMTagkJdBX2z7Il/jGhDeJ3j6s80
x-amz-request-id: 32FE2CEB32F5EE25
Date: Wed, 01 Mar 2006 12:00:00 GMT
Connection: close
Server: AmazonS3

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3
DeleteBucketAnalyticsConfiguration
Service: Amazon Simple Storage Service

Deletes an analytics configuration for the bucket (specified by the analytics configuration ID).

To use this operation, you must have permissions to perform the s3:PutAnalyticsConfiguration action. The bucket owner has this permission by default. The bucket owner can grant this permission to others. For more information about permissions, see Permissions Related to Bucket Subresource Operations and Managing Access Permissions to Your Amazon S3 Resources.

For information about the Amazon S3 analytics feature, see Amazon S3 Analytics – Storage Class Analysis.

The following operations are related to DeleteBucketAnalyticsConfiguration:

- GetBucketAnalyticsConfiguration
- ListBucketAnalyticsConfigurations
- PutBucketAnalyticsConfiguration

Request Syntax

```
DELETE /?analytics&id=Id HTTP/1.1
Host: Bucket.s3.amazonaws.com
x-amz-expected-bucket-owner: ExpectedBucketOwner
```

URI Request Parameters

The request uses the following URI parameters.

**Bucket (p. 45)**

The name of the bucket from which an analytics configuration is deleted.

Required: Yes

**id (p. 45)**

The ID that identifies the analytics configuration.

Required: Yes

**x-amz-expected-bucket-owner (p. 45)**

The account id of the expected bucket owner. If the bucket is owned by a different account, the request will fail with an HTTP 403 (Access Denied) error.

Request Body

The request does not have a request body.

Response Syntax

```
HTTP/1.1 204
```

Response Elements

If the action is successful, the service sends back an HTTP 204 response with an empty HTTP body.
Examples

Sample Request

The following DELETE request deletes the analytics configuration with the ID list1.

```
DELETE ?/analytics&id=list1 HTTP/1.1
Host: examplebucket.s3.<Region>.amazonaws.com
Date: Wed, 14 May 2014 02:11:22 GMT
Authorization: signatureValue
```

Sample Response

The following successful response shows Amazon S3 returning a 204 No Content response. The analytics configuration with the ID list1 for the bucket has been removed.

```
HTTP/1.1 204 No Content
x-amz-id-2: 0FmFIWsh/PpBuzZ0JFRC55ZGVmQW4SHJ7xVDqKwhEdJmf3g63RtrvH8ZuxW1Bo15
x-amz-request-id: 0CF038E9BCF63097
Date: Wed, 14 May 2014 02:11:22 GMT
Server: AmazonS3
```

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3
DeleteBucketCors
Service: Amazon Simple Storage Service

Deletes the cors configuration information set for the bucket.

To use this operation, you must have permission to perform the s3:PutBucketCORS action. The bucket owner has this permission by default and can grant this permission to others.

For information about cors, see Enabling Cross-Origin Resource Sharing in the Amazon Simple Storage Service Developer Guide.

Related Resources:
- PutBucketCors
- RESTOPTIONSobject

Request Syntax

```
DELETE /?cors HTTP/1.1
Host: Bucket.s3.amazonaws.com
x-amz-expected-bucket-owner: ExpectedBucketOwner
```

URI Request Parameters

The request uses the following URI parameters.

**Bucket (p. 47)**

Specifies the bucket whose cors configuration is being deleted.

Required: Yes

**x-amz-expected-bucket-owner (p. 47)**

The account id of the expected bucket owner. If the bucket is owned by a different account, the request will fail with an HTTP 403 (Access Denied) error.

Request Body

The request does not have a request body.

Response Syntax

```
HTTP/1.1 204
```

Response Elements

If the action is successful, the service sends back an HTTP 204 response with an empty HTTP body.

Examples

Retrieve cors subresource

The following DELETE request deletes the cors subresource from the specified bucket. This action removes cors configuration that is stored in the subresource.
Sample Request

This example illustrates one usage of DeleteBucketCors.

DELETE /?cors HTTP/1.1
Host: examplebucket.s3.<Region>.amazonaws.com
Date: Tue, 13 Dec 2011 19:14:42 GMT
Authorization: signatureValue

Sample Response

This example illustrates one usage of DeleteBucketCors.

DELETE /?cors HTTP/1.1
Host: examplebucket.s3.<Region>.amazonaws.com
Date: Tue, 13 Dec 2011 19:14:42 GMT
Authorization: signatureValue

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3
DeleteBucketEncryption
Service: Amazon Simple Storage Service

This implementation of the DELETE operation removes default encryption from the bucket. For information about the Amazon S3 default encryption feature, see Amazon S3 Default Bucket Encryption in the Amazon Simple Storage Service Developer Guide.

To use this operation, you must have permissions to perform the s3:PutEncryptionConfiguration action. The bucket owner has this permission by default. The bucket owner can grant this permission to others. For more information about permissions, see Permissions Related to Bucket Subresource Operations and Managing Access Permissions to your Amazon S3 Resources in the Amazon Simple Storage Service Developer Guide.

Related Resources
• PutBucketEncryption
• GetBucketEncryption

Request Syntax

```plaintext
DELETE /?encryption HTTP/1.1
Host: Bucket.s3.amazonaws.com
x-amz-expected-bucket-owner: ExpectedBucketOwner
```

URI Request Parameters

The request uses the following URI parameters.

Bucket (p. 49)

The name of the bucket containing the server-side encryption configuration to delete.

Required: Yes

x-amz-expected-bucket-owner (p. 49)

The account id of the expected bucket owner. If the bucket is owned by a different account, the request will fail with an HTTP 403 (Access Denied) error.

Request Body

The request does not have a request body.

Response Syntax

```plaintext
HTTP/1.1 204
```

Response Elements

If the action is successful, the service sends back an HTTP 204 response with an empty HTTP body.

Examples

Sample Request

The following DELETE request deletes default encryption from the bucket.
Sample Response

The following successful response shows Amazon S3 returning a 204 No Content response confirming that default encryption has been removed from the bucket.

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3
DeleteBucketIntelligentTieringConfiguration
Service: Amazon Simple Storage Service

Deletes the S3 Intelligent-Tiering configuration from the specified bucket.

The S3 Intelligent-Tiering storage class is designed to optimize storage costs by automatically moving data to the most cost-effective storage access tier, without additional operational overhead. S3 Intelligent-Tiering delivers automatic cost savings by moving data between access tiers, when access patterns change.

The S3 Intelligent-Tiering storage class is suitable for objects larger than 128 KB that you plan to store for at least 30 days. If the size of an object is less than 128 KB, it is not eligible for auto-tiering. Smaller objects can be stored, but they are always charged at the frequent access tier rates in the S3 Intelligent-Tiering storage class.

If you delete an object before the end of the 30-day minimum storage duration period, you are charged for 30 days. For more information, see Storage class for automatically optimizing frequently and infrequently accessed objects.

Operations related to DeleteBucketIntelligentTieringConfiguration include:

- GetBucketIntelligentTieringConfiguration
- PutBucketIntelligentTieringConfiguration
- ListBucketIntelligentTieringConfigurations

Request Syntax

```
DELETE /?intelligent-tiering&id=Id HTTP/1.1
Host: Bucket.s3.amazonaws.com
```

URI Request Parameters

The request uses the following URI parameters.

**Bucket (p. 51)**

The name of the Amazon S3 bucket whose configuration you want to modify or retrieve.

Required: Yes

**id (p. 51)**

The ID used to identify the S3 Intelligent-Tiering configuration.

Required: Yes

Request Body

The request does not have a request body.

Response Syntax

```
HTTP/1.1 204
```

Response Elements

If the action is successful, the service sends back an HTTP 204 response with an empty HTTP body.
See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3
DeleteBucketInventoryConfiguration
Service: Amazon Simple Storage Service

Deletes an inventory configuration (identified by the inventory ID) from the bucket.

To use this operation, you must have permissions to perform the s3:PutInventoryConfiguration action. The bucket owner has this permission by default. The bucket owner can grant this permission to others. For more information about permissions, see Permissions Related to Bucket Subresource Operations and Managing Access Permissions to Your Amazon S3 Resources.

For information about the Amazon S3 inventory feature, see Amazon S3 Inventory.

Operations related to DeleteBucketInventoryConfiguration include:

- GetBucketInventoryConfiguration
- PutBucketInventoryConfiguration
- ListBucketInventoryConfigurations

Request Syntax

```
DELETE /?inventory&id=Id HTTP/1.1
Host: Bucket.s3.amazonaws.com
x-amz-expected-bucket-owner: ExpectedBucketOwner
```

URI Request Parameters

The request uses the following URI parameters.

Bucket (p. 53)

The name of the bucket containing the inventory configuration to delete.

Required: Yes

id (p. 53)

The ID used to identify the inventory configuration.

Required: Yes

x-amz-expected-bucket-owner (p. 53)

The account id of the expected bucket owner. If the bucket is owned by a different account, the request will fail with an HTTP 403 (Access Denied) error.

Request Body

The request does not have a request body.

Response Syntax

```
HTTP/1.1 204
```

Response Elements

If the action is successful, the service sends back an HTTP 204 response with an empty HTTP body.
Examples

Sample Request

The following DELETE request deletes the inventory configuration with the ID list1.

```
DELETE /inventory&id=list1 HTTP/1.1
Host: examplebucket.s3.<Region>.amazonaws.com
Date: Wed, 14 May 2014 02:11:22 GMT
Authorization: signatureValue
```

Sample Response

The following successful response shows Amazon S3 returning a 204 No Content response. The inventory configuration with the ID list1 for the bucket has been removed.

```
HTTP/1.1 204 No Content
x-amz-id-2: 0FmFIWsh/PpBuzZ0JFRC55ZGVMQW4SHJ7xVDqKWhEd3mf3q63RtrvH8ZuxW1Bo15
x-amz-request-id: 0CF038E9BC63097
Date: Wed, 14 May 2014 02:11:22 GMT
Server: AmazonS3
```

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3
DeleteBucketLifecycle
Service: Amazon Simple Storage Service

Deletes the lifecycle configuration from the specified bucket. Amazon S3 removes all the lifecycle configuration rules in the lifecycle subresource associated with the bucket. Your objects never expire, and Amazon S3 no longer automatically deletes any objects on the basis of rules contained in the deleted lifecycle configuration.

To use this operation, you must have permission to perform the s3:PutLifecycleConfiguration action. By default, the bucket owner has this permission and the bucket owner can grant this permission to others.

There is usually some time lag before lifecycle configuration deletion is fully propagated to all the Amazon S3 systems.

For more information about the object expiration, see Elements to Describe Lifecycle Actions.

Related actions include:
• PutBucketLifecycleConfiguration
• GetBucketLifecycleConfiguration

Request Syntax

DELETE /?lifecycle HTTP/1.1
Host: Bucket.s3.amazonaws.com
x-amz-expected-bucket-owner: ExpectedBucketOwner

URI Request Parameters

The request uses the following URI parameters.

Bucket (p. 55)

The bucket name of the lifecycle to delete.

Required: Yes

x-amz-expected-bucket-owner (p. 55)

The account id of the expected bucket owner. If the bucket is owned by a different account, the request will fail with an HTTP 403 (Access Denied) error.

Request Body

The request does not have a request body.

Response Syntax

HTTP/1.1 204

Response Elements

If the action is successful, the service sends back an HTTP 204 response with an empty HTTP body.
Examples

Sample Request

The following DELETE request deletes the lifecycle subresource from the specified bucket. This removes lifecycle configuration stored in the subresource.

```
DELETE /?lifecycle HTTP/1.1
Host: examplebucket.s3.<Region>.amazonaws.com
Date: Wed, 14 Dec 2011 05:37:16 GMT
Authorization: signatureValue
```

Sample Response

The following successful response shows Amazon S3 returning a 204 No Content response. Objects in your bucket no longer expire.

```
HTTP/1.1 204 No Content
x-amz-id-2: Uuaq1LuByRlx9e6j5OnimrSAMPLEtRFyTaOAa==
x-amz-request-id: 656c76696e672SAMPLE5657374
Date: Wed, 14 Dec 2011 05:37:16 GMT
Connection: keep-alive
Server: AmazonS3
```

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3
DeleteBucketMetricsConfiguration
Service: Amazon Simple Storage Service

Deletes a metrics configuration for the Amazon CloudWatch request metrics (specified by the metrics configuration ID) from the bucket. Note that this doesn't include the daily storage metrics.

To use this operation, you must have permissions to perform the `s3:PutMetricsConfiguration` action. The bucket owner has this permission by default. The bucket owner can grant this permission to others. For more information about permissions, see Permissions Related to Bucket Subresource Operations and Managing Access Permissions to Your Amazon S3 Resources.

For information about CloudWatch request metrics for Amazon S3, see Monitoring Metrics with Amazon CloudWatch.

The following operations are related to DeleteBucketMetricsConfiguration:

- GetBucketMetricsConfiguration
- PutBucketMetricsConfiguration
- ListBucketMetricsConfigurations
- Monitoring Metrics with Amazon CloudWatch

Request Syntax

```
DELETE /?metrics&id=Id HTTP/1.1
Host: Bucket.s3.amazonaws.com
x-amz-expected-bucket-owner: ExpectedBucketOwner
```

URI Request Parameters

The request uses the following URI parameters.

**Bucket (p. 57)**

The name of the bucket containing the metrics configuration to delete.

Required: Yes

**id (p. 57)**

The ID used to identify the metrics configuration.

Required: Yes

**x-amz-expected-bucket-owner (p. 57)**

The account ID of the expected bucket owner. If the bucket is owned by a different account, the request will fail with an HTTP 403 (Access Denied) error.

Request Body

The request does not have a request body.

Response Syntax

```
HTTP/1.1 204
```
Response Elements

If the action is successful, the service sends back an HTTP 204 response with an empty HTTP body.

Examples

Sample Request

Delete the metric configuration with a specified ID, which disables the CloudWatch metrics with the ExampleMetrics value for the FilterId dimension.

```
DELETE /?metrics&id=ExampleMetrics HTTP/1.1
Host: examplebucket.s3.<Region>.amazonaws.com
x-amz-date: Thu, 15 Nov 2016 00:17:21 GMT
Authorization: signatureValue
```

Sample Response

Delete the metric configuration with a specified ID, which disables the CloudWatch metrics with the ExampleMetrics value for the FilterId dimension.

```
HTTP/1.1 204 No Content
x-amz-id-2: ITnGT1y4REXAMPLEPi4hklTXouTf0hccUjo0iCPEXAMPLEutBj3M7fPGlW02SEWp
x-amz-request-id: 51991EXAMPLE5321
Date: Thu, 15 Nov 2016 00:17:22 GMT
Server: AmazonS3
```

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3
DeleteBucketOwnershipControls
Service: Amazon Simple Storage Service

Removes OwnershipControls for an Amazon S3 bucket. To use this operation, you must have the s3:PutBucketOwnershipControls permission. For more information about Amazon S3 permissions, see Specifying Permissions in a Policy.

For information about Amazon S3 Object Ownership, see Using Object Ownership.

The following operations are related to DeleteBucketOwnershipControls:

- GetBucketOwnershipControls (p. 131)
- PutBucketOwnershipControls (p. 314)

Request Syntax

```
DELETE /?ownershipControls HTTP/1.1
Host: Bucket.s3.amazonaws.com
x-amz-expected-bucket-owner: ExpectedBucketOwner
```

URI Request Parameters

The request uses the following URI parameters.

**Bucket (p. 59)**

The Amazon S3 bucket whose OwnershipControls you want to delete.

Required: Yes

**x-amz-expected-bucket-owner (p. 59)**

The account id of the expected bucket owner. If the bucket is owned by a different account, the request will fail with an HTTP 403 (Access Denied) error.

Request Body

The request does not have a request body.

Response Syntax

```
HTTP/1.1 204
```

Response Elements

If the action is successful, the service sends back an HTTP 204 response with an empty HTTP body.

Examples

Sample DeleteBucketOwnershipControls Request

This example illustrates one usage of DeleteBucketOwnershipControls.

```
DELETE /example-bucket?/ownershipControls HTTP/1.1
Host: examplebucket.s3.<Region>.amazonaws.com
```
Sample DeleteBucketOwnershipControls Response

This example illustrates one usage of DeleteBucketOwnershipControls.

HTTP/1.1 204 No Content
x-amz-id-2: dVrxJD3XHDcjjZHFtd7eSB+ovpY8hQ6kSe9jpzyRVkwp27cij05qV1pT1vz/
hjlsrupiy9gEkSdw=
x-amz-request-id: 4BF0B777B446C97
Date: Thu, 18 Jun 2020 22:54:03 GMT
Server: AmazonS3

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3
DeleteBucketPolicy
Service: Amazon Simple Storage Service

This implementation of the DELETE operation uses the policy subresource to delete the policy of a specified bucket. If you are using an identity other than the root user of the AWS account that owns the bucket, the calling identity must have the DeleteBucketPolicy permissions on the specified bucket and belong to the bucket owner's account to use this operation.

If you don't have DeleteBucketPolicy permissions, Amazon S3 returns a 403 Access Denied error. If you have the correct permissions, but you're not using an identity that belongs to the bucket owner's account, Amazon S3 returns a 405 Method Not Allowed error.

Important
As a security precaution, the root user of the AWS account that owns a bucket can always use this operation, even if the policy explicitly denies the root user the ability to perform this action.

For more information about bucket policies, see Using Bucket Policies and UserPolicies.

The following operations are related to DeleteBucketPolicy
- CreateBucket
- DeleteObject

Request Syntax
DELETE /?policy HTTP/1.1
Host: Bucket.s3.amazonaws.com
x-amz-expected-bucket-owner: ExpectedBucketOwner

URI Request Parameters
The request uses the following URI parameters.

Bucket (p. 61)
The bucket name.
Required: Yes

x-amz-expected-bucket-owner (p. 61)
The account id of the expected bucket owner. If the bucket is owned by a different account, the request will fail with an HTTP 403 (Access Denied) error.

Request Body
The request does not have a request body.

Response Syntax
HTTP/1.1 204

Response Elements
If the action is successful, the service sends back an HTTP 204 response with an empty HTTP body.
Examples

Sample Request

This request deletes the bucket named BucketName.

```
DELETE /?policy HTTP/1.1
Host: BucketName.s3.<Region>.amazonaws.com
Date: Tue, 04 Apr 2010 20:34:56 GMT
Authorization: signatureValue
```

Sample Response

This example illustrates one usage of DeleteBucketPolicy.

```
HTTP/1.1 204 No Content
x-amz-id-2: Uuag1LuByRx9e6j5OnimrSAMPLEtRpfTaOFg==
x-amz-request-id: 656c76696e672SAMPLE5657374
Date: Tue, 04 Apr 2010 20:34:56 GMT
Connection: keep-alive
Server: AmazonS3
```

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3
DeleteBucketReplication
Service: Amazon Simple Storage Service

Deletes the replication configuration from the bucket.

To use this operation, you must have permissions to perform the s3:PutReplicationConfiguration action. The bucket owner has these permissions by default and can grant it to others. For more information about permissions, see Permissions Related to Bucket Subresource Operations and Managing Access Permissions to Your Amazon S3 Resources.

Note
It can take a while for the deletion of a replication configuration to fully propagate.

For information about replication configuration, see Replication in the Amazon S3 Developer Guide.

The following operations are related to DeleteBucketReplication:

- PutBucketReplication
- GetBucketReplication

Request Syntax

```
DELETE /?replication HTTP/1.1
Host: Bucket.s3.amazonaws.com
x-amz-expected-bucket-owner: ExpectedBucketOwner
```

URI Request Parameters

The request uses the following URI parameters.

**Bucket (p. 63)**

The bucket name.

Required: Yes

**x-amz-expected-bucket-owner (p. 63)**

The account id of the expected bucket owner. If the bucket is owned by a different account, the request will fail with an HTTP 403 (Access Denied) error.

Request Body

The request does not have a request body.

Response Syntax

```
HTTP/1.1 204
```

Response Elements

If the action is successful, the service sends back an HTTP 204 response with an empty HTTP body.

Examples

Sample Request

The following DELETE request deletes the replication subresource from the specified bucket. This removes the replication configuration that is set for the bucket.
DELETE /?replication HTTP/1.1
Host: examplebucket.s3.<Region>.amazonaws.com
Date: Wed, 11 Feb 2015 05:37:16 GMT
20150211T171320Z
Authorization: authorization string

Sample Response

When the replication subresource has been deleted, Amazon S3 returns a 204 No Content response. It will not replicate new objects that are stored in the examplebucket bucket.

HTTP/1.1 204 No Content
x-amz-id-2: Uuag1LuByRx9e6j5OnimrSAMPLEtRPfTaOa==
x-amz-request-id: 656c76696e672example
Date: Wed, 11 Feb 2015 05:37:16 GMT
Connection: keep-alive
Server: AmazonS3

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3
DeleteBucketTagging

Service: Amazon Simple Storage Service

Deletes the tags from the bucket.

To use this operation, you must have permission to perform the s3:PutBucketTagging action. By default, the bucket owner has this permission and can grant this permission to others.

The following operations are related to DeleteBucketTagging:

- GetBucketTagging
- PutBucketTagging

Request Syntax

```
DELETE /?tagging HTTP/1.1
Host: Bucket.s3.amazonaws.com
x-amz-expected-bucket-owner: ExpectedBucketOwner
```

URI Request Parameters

The request uses the following URI parameters.

**Bucket (p. 65)**

The bucket that has the tag set to be removed.

Required: Yes

**x-amz-expected-bucket-owner (p. 65)**

The account id of the expected bucket owner. If the bucket is owned by a different account, the request will fail with an HTTP 403 (Access Denied) error.

Request Body

The request does not have a request body.

Response Syntax

```
HTTP/1.1 204
```

Response Elements

If the action is successful, the service sends back an HTTP 204 response with an empty HTTP body.

Examples

Sample Request

The following DELETE request deletes the tag set from the specified bucket.

```
DELETE /?tagging HTTP/1.1
Host: examplebucket.s3.<Region>.amazonaws.com
Date: Wed, 14 Dec 2011 05:37:16 GMT
```
Authorization: signatureValue

Sample Response

The following successful response shows Amazon S3 returning a 204 No Content response. The tag set for the bucket has been removed.

HTTP/1.1 204 No Content
Date: Wed, 25 Nov 2009 12:00:00 GMT
Connection: close
Server: AmazonS3

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3
DeleteBucketWebsite

Service: Amazon Simple Storage Service

This operation removes the website configuration for a bucket. Amazon S3 returns a 200 OK response upon successfully deleting a website configuration on the specified bucket. You will get a 200 OK response if the website configuration you are trying to delete does not exist on the bucket. Amazon S3 returns a 404 response if the bucket specified in the request does not exist.

This DELETE operation requires the S3:DeleteBucketWebsite permission. By default, only the bucket owner can delete the website configuration attached to a bucket. However, bucket owners can grant other users permission to delete the website configuration by writing a bucket policy granting them the S3:DeleteBucketWebsite permission.

For more information about hosting websites, see Hosting Websites on Amazon S3.

The following operations are related to DeleteBucketWebsite:

- GetBucketWebsite
- PutBucketWebsite

Request Syntax

DELETE /?website HTTP/1.1
Host: Bucket.s3.amazonaws.com
x-amz-expected-bucket-owner: ExpectedBucketOwner

URI Request Parameters

The request uses the following URI parameters.

Bucket (p. 67)

The bucket name for which you want to remove the website configuration.

Required: Yes

x-amz-expected-bucket-owner (p. 67)

The account id of the expected bucket owner. If the bucket is owned by a different account, the request will fail with an HTTP 403 (Access Denied) error.

Request Body

The request does not have a request body.

Response Syntax

HTTP/1.1 204

Response Elements

If the action is successful, the service sends back an HTTP 204 response with an empty HTTP body.

Examples

Sample Request

This request deletes the website configuration on the specified bucket.
DELETE /website HTTP/1.1
Host: example-bucket.s3.<Region>.amazonaws.com
Date: Thu, 27 Jan 2011 12:00:00 GMT
Authorization: signatureValue

Sample Response

This example illustrates one usage of DeleteBucketWebsite.

HTTP/1.1 204 No Content
x-amz-id-2: aws-s3integ-s3ws-31008.sea31.amazon.com
x-amz-request-id: AF1DD829D3B49707
Date: Thu, 03 Feb 2011 22:10:26 GMT
Server: AmazonS3

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3
DeleteObject
Service: Amazon Simple Storage Service

Removes the null version (if there is one) of an object and inserts a delete marker, which becomes the latest version of the object. If there isn't a null version, Amazon S3 does not remove any objects.

To remove a specific version, you must be the bucket owner and you must use the version Id subresource. Using this subresource permanently deletes the version. If the object deleted is a delete marker, Amazon S3 sets the response header, x-amz-delete-marker, to true.

If the object you want to delete is in a bucket where the bucket versioning configuration is MFA Delete enabled, you must include the x-amz-mfa request header in the DELETE versionId request. Requests that include x-amz-mfa must use HTTPS.

For more information about MFA Delete, see Using MFA Delete. To see sample requests that use versioning, see Sample Request.

You can delete objects by explicitly calling the DELETE Object API or configure its lifecycle (PutBucketLifecycle) to enable Amazon S3 to remove them for you. If you want to block users or accounts from removing or deleting objects from your bucket, you must deny them the s3:DeleteObject, s3:DeleteObjectVersion, and s3:PutLifeCycleConfiguration actions.

The following operation is related to DeleteObject:
• PutObject

Request Syntax

DELETE /Key+?versionId=VersionId HTTP/1.1
Host: Bucket.s3.amazonaws.com
x-amz-mfa: MFA
x-amz-request-payer: RequestPayer
x-amz-bypass-governance-retention: BypassGovernanceRetention
x-amz-expected-bucket-owner: ExpectedBucketOwner

URI Request Parameters

The request uses the following URI parameters.

Bucket (p. 69)

The bucket name of the bucket containing the object.

When using this API with an access point, you must direct requests to the access point hostname. The access point hostname takes the form AccessPointName-AccountId.s3-accesspoint.Region.amazonaws.com. When using this operation with an access point through the AWS SDKs, you provide the access point ARN in place of the bucket name. For more information about access point ARNs, see Using Access Points in the Amazon Simple Storage Service Developer Guide.

When using this API with Amazon S3 on Outposts, you must direct requests to the S3 on Outposts hostname. The S3 on Outposts hostname takes the form AccessPointName-AccountId.outpostID.s3-outposts.Region.amazonaws.com. When using this operation using S3 on Outposts through the AWS SDKs, you provide the Outposts bucket ARN in place of the bucket name. For more information about S3 on Outposts ARNs, see Using S3 on Outposts in the Amazon Simple Storage Service Developer Guide.

Required: Yes
Key (p. 69)

Key name of the object to delete.

Length Constraints: Minimum length of 1.

Required: Yes

versionId (p. 69)

VersionId used to reference a specific version of the object.

x-amz-bypass-governance-retention (p. 69)

Indicates whether S3 Object Lock should bypass Governance-mode restrictions to process this operation.

x-amz-expected-bucket-owner (p. 69)

The account id of the expected bucket owner. If the bucket is owned by a different account, the request will fail with an HTTP 403 (Access Denied) error.

x-amz-mfa (p. 69)

The concatenation of the authentication device's serial number, a space, and the value that is displayed on your authentication device. Required to permanently delete a versioned object if versioning is configured with MFA delete enabled.

x-amz-request-payer (p. 69)

Confirms that the requester knows that they will be charged for the request. Bucket owners need not specify this parameter in their requests. For information about downloading objects from requester pays buckets, see Downloading Objects in Requestor Pays Buckets in the Amazon S3 Developer Guide.

Valid Values: requester

Request Body

The request does not have a request body.

Response Syntax

HTTP/1.1 204
x-amz-delete-marker: DeleteMarker
x-amz-version-id: VersionId
x-amz-request-charged: RequestCharged

Response Elements

If the action is successful, the service sends back an HTTP 204 response.

The response returns the following HTTP headers.

x-amz-delete-marker (p. 70)

Specifies whether the versioned object that was permanently deleted was (true) or was not (false) a delete marker.

x-amz-request-charged (p. 70)

If present, indicates that the requester was successfully charged for the request.
Valid Values: requester

x-amz-version-id (p. 70)

Returns the version ID of the delete marker created as a result of the DELETE operation.

Examples

Sample Request

The following request deletes the object my-second-image.jpg.

```
DELETE /my-second-image.jpg HTTP/1.1
Host: bucket.s3.<Region>.amazonaws.com
Date: Wed, 12 Oct 2009 17:50:00 GMT
Authorization: authorization string
Content-Type: text/plain
```

Sample Response

This example illustrates one usage of DeleteObject.

```
HTTP/1.1 204 NoContent
x-amz-id-2: LriYPLdmOdA1fgSm/F1YsViT1LW94/xUQxMsF7xiEblaOwIIOId+zbw2l63pt7
x-amz-request-id: 0A49CE4060975EAC
Date: Wed, 12 Oct 2009 17:50:00 GMT
Content-Length: 0
Connection: close
Server: AmazonS3
```

Sample Request: Deleting a specified version of an object

The following request deletes the specified version of the object my-third-image.jpg.

```
DELETE /my-third-image.jpg?versionId=UIORU8nfdIufdiosojhr398493jfdkFJjkn9nqUifhwnw89493jJFJ HTTP/1.1
Host: bucket.s3.<Region>.amazonaws.com
Date: Wed, 12 Oct 2009 17:50:00 GMT
Authorization: authorization string
Content-Type: text/plain
Content-Length: 0
```

Sample Response

This example illustrates one usage of DeleteObject.

```
HTTP/1.1 204 NoContent
x-amz-id-2: LriYPLdmOdA1fgSm/F1YsViT1LW94/xUQxMsF7xiEblaOwIIOId+zbw2l63pt7
x-amz-request-id: 0A49CE4060975EAC
x-amz-version-id: UIORU8nfdIufdiosojhr398493jfdkFJjkn9nqUifhwnw89493jJFJ
Date: Wed, 12 Oct 2009 17:50:00 GMT
Content-Length: 0
```
Sample Response: If the object deleted is a delete marker

This example illustrates one usage of DeleteObject.

```
HTTP/1.1 204 NoContent
x-amz-id-2: LrijVLdmOdA1lfgSm/F1YsViT1LW94/xUQxMsF7xiEbla0wi1O1x1+z2bwZ163pt7
x-amz-request-id: 0A49CE4060975EAC
x-amz-version-id: 3/L4kqtJlpXgDRD7mJ+rmSpXv3dIbrHY
+MTRCxf3vJVBH408x8xq0PUMLuo
x-amz-delete-marker: true
Date: Wed, 12 Oct 2009 17:50:00 GMT
Content-Length: 0
Connection: close
Server: AmazonS3
```

Sample Request: Deleting a specified version of an object in an MFA-enabled bucket

The following request deletes the specified version of the object my-third-image.jpg, which is stored in an MFA-enabled bucket.

```
DELETE /my-third-image.jpg?versionId=UIORUnfndlfuf HTTP/1.1
Host: bucket.s3.<Region>.amazonaws.com
Date: Wed, 12 Oct 2009 17:50:00 GMT
x-amz-mfa:[SerialNumber] [AuthenticationCode]
Authorization: authorization string
Content-Type: text/plain
Content-Length: 0
```

Sample Response

This example illustrates one usage of DeleteObject.

```
HTTP/1.1 204 NoContent
x-amz-id-2: LrijVLdmOdA1lfgSm/F1YsViT1LW94/xUQxMsF7xiEbla0wi1O1x1+z2bwZ163pt7
x-amz-request-id: 0A49CE4060975EAC
x-amz-version-id: UIORUnfndlfuf
Date: Wed, 12 Oct 2009 17:50:00 GMT
Content-Length: 0
Connection: close
Server: AmazonS3
```

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
• AWS SDK for Java
• AWS SDK for JavaScript
• AWS SDK for PHP V3
• AWS SDK for Python
• AWS SDK for Ruby V3
DeleteObjects
Service: Amazon Simple Storage Service

This operation enables you to delete multiple objects from a bucket using a single HTTP request. If you know the object keys that you want to delete, then this operation provides a suitable alternative to sending individual delete requests, reducing per-request overhead.

The request contains a list of up to 1000 keys that you want to delete. In the XML, you provide the object key names, and optionally, version IDs if you want to delete a specific version of the object from a versioning-enabled bucket. For each key, Amazon S3 performs a delete operation and returns the result of that delete, success, or failure, in the response. Note that if the object specified in the request is not found, Amazon S3 returns the result as deleted.

The operation supports two modes for the response: verbose and quiet. By default, the operation uses verbose mode in which the response includes the result of deletion of each key in your request. In quiet mode the response includes only keys where the delete operation encountered an error. For a successful deletion, the operation does not return any information about the delete in the response body.

When performing this operation on an MFA Delete enabled bucket, that attempts to delete any versioned objects, you must include an MFA token. If you do not provide one, the entire request will fail, even if there are non-versioned objects you are trying to delete. If you provide an invalid token, whether there are versioned keys in the request or not, the entire Multi-Object Delete request will fail. For information about MFA Delete, see MFA Delete.

Finally, the Content-MD5 header is required for all Multi-Object Delete requests. Amazon S3 uses the header value to ensure that your request body has not been altered in transit.

The following operations are related to DeleteObjects:
- CreateMultipartUpload
- UploadPart
- CompleteMultipartUpload
- ListParts
- AbortMultipartUpload

Request Syntax

```plaintext
POST /?delete HTTP/1.1
Host: Bucket.s3.amazonaws.com
x-amz-mfa: MFA
x-amz-request-payer: RequestPayer
x-amz-bypass-governance-retention: BypassGovernanceRetention
x-amz-expected-bucket-owner: ExpectedBucketOwner
<?xml version="1.0" encoding="UTF-8"?>
<Delete xmlns="http://s3.amazonaws.com/doc/2006-03-01/">
  <Object>
    <Key>string</Key>
    <VersionId>string</VersionId>
  </Object>
  ...
  <Quiet>boolean</Quiet>
</Delete>
```

URI Request Parameters

The request uses the following URI parameters.
Bucket (p. 74)

The bucket name containing the objects to delete.

When using this API with an access point, you must direct requests to the access point hostname. The access point hostname takes the form AccessPointName-AccountId.s3-accesspoint.Region.amazonaws.com. When using this operation with an access point through the AWS SDKs, you provide the access point ARN in place of the bucket name. For more information about access point ARNs, see Using Access Points in the Amazon Simple Storage Service Developer Guide.

When using this API with Amazon S3 on Outposts, you must direct requests to the S3 on Outposts hostname. The S3 on Outposts hostname takes the form AccessPointName-AccountId.outpostId.s3-outposts.Region.amazonaws.com. When using this operation using S3 on Outposts through the AWS SDKs, you provide the Outposts bucket ARN in place of the bucket name. For more information about S3 on Outposts ARNs, see Using S3 on Outposts in the Amazon Simple Storage Service Developer Guide.

Required: Yes

x-amz-bypass-governance-retention (p. 74)

Specifies whether you want to delete this object even if it has a Governance-type Object Lock in place. You must have sufficient permissions to perform this operation.

x-amz-expected-bucket-owner (p. 74)

The account id of the expected bucket owner. If the bucket is owned by a different account, the request will fail with an HTTP 403 (Access Denied) error.

x-amz-mfa (p. 74)

The concatenation of the authentication device's serial number, a space, and the value that is displayed on your authentication device. Required to permanently delete a versioned object if versioning is configured with MFA delete enabled.

x-amz-request-payer (p. 74)

Confirms that the requester knows that they will be charged for the request. Bucket owners need not specify this parameter in their requests. For information about downloading objects from requester pays buckets, see Downloading Objects in Requestor Pays Buckets in the Amazon S3 Developer Guide.

Valid Values: requester

Request Body

The request accepts the following data in XML format.

Delete (p. 74)

Root level tag for the Delete parameters.

Required: Yes

Object (p. 74)

The objects to delete.

Type: Array of ObjectIdentifier (p. 608) data types

Required: Yes
Quiet (p. 74)

Element to enable quiet mode for the request. When you add this element, you must set its value to true.

Type: Boolean

Required: No

Response Syntax

HTTP/1.1 200
x-amz-request-charged: RequestCharged
<?xml version="1.0" encoding="UTF-8"?>
<DeleteResult>
  <Deleted>
    <DeleteMarker>boolean</DeleteMarker>
    <DeleteMarkerVersionId>string</DeleteMarkerVersionId>
    <Key>string</Key>
    <VersionId>string</VersionId>
  </Deleted>
  ...
  <Error>
    <Code>string</Code>
    <Key>string</Key>
    <Message>string</Message>
    <VersionId>string</VersionId>
  </Error>
  ...
</DeleteResult>

Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The response returns the following HTTP headers.

x-amz-request-charged (p. 76)

If present, indicates that the requester was successfully charged for the request.

Valid Values: requester

The following data is returned in XML format by the service.

DeleteResult (p. 76)

Root level tag for the DeleteResult parameters.

Required: Yes

Deleted (p. 76)

Container element for a successful delete. It identifies the object that was successfully deleted.

Type: Array of DeletedObject (p. 544) data types

Error (p. 76)

Container for a failed delete operation that describes the object that Amazon S3 attempted to delete and the error it encountered.

Type: Array of Error (p. 552) data types
Examples

Sample Request: Multi-object delete resulting in mixed success/error response

This example illustrates a Multi-Object Delete request to delete objects that result in mixed success and errors response. The following request deletes two objects from a bucket (bucketname). In this example, the requester does not have permission to delete the sample2.txt object.

```
POST /?delete HTTP/1.1
Host: bucketname.s3.<Region>.amazonaws.com
Accept: */*
x-amz-date: Wed, 30 Nov 2011 03:39:05 GMT
Content-MD5: p5/WA/oEr30qrEEl21PAqw==
Authorization: AWS AKIAIOSFODNN7EXAMPLE:W0qPYCLe6JwkZAD1ei6hp9XZIee=
Content-Length: 125
Connection: Keep-Alive

<Delete>
  <Object>
    <Key>sample1.txt</Key>
  </Object>
  <Object>
    <Key>sample2.txt</Key>
  </Object>
</Delete>
```

Sample Response

The response includes a DeleteResult element that includes a Deleted element for the item that Amazon S3 successfully deleted and an Error element that Amazon S3 did not delete because you didn't have permission to delete the object.

```
HTTP/1.1 200 OK
x-amz-id-2: 5h4FxSNCUS7wP5z92eGCWDshNpMnRuXvETa4HH3LvvH6VAIr0jU7tH9kM7X+njXx
x-amz-request-id: A43783B641629AEE
Date: Fri, 02 Dec 2011 01:53:42 GMT
Content-Type: application/xml
Server: AmazonS3
Content-Length: 251

<?xml version="1.0" encoding="UTF-8"?>
<DeleteResult xmlns="http://s3.amazonaws.com/doc/2006-03-01/">
  <Deleted>
    <Key>sample1.txt</Key>
  </Deleted>
  <Error>
    <Key>sample2.txt</Key>
    <Code>AccessDenied</Code>
    <Message>Access Denied</Message>
  </Error>
</DeleteResult>
```

Sample Request: Deleting an object from a versioned bucket

If you delete an item from a versioning enabled bucket, all versions of that object remain in the bucket; however, Amazon S3 inserts a delete marker. For more information, see Object Versioning.
The following scenarios describe the behavior of a multi-object Delete request when versioning is enabled for your bucket.

Case 1 - Simple Delete: In the following sample request, the multi-object delete request specifies only one key.

```xml
POST /delete HTTP/1.1
Host: bucketname.s3.<Region>.amazonaws.com
Accept: */*
x-amz-date: Wed, 30 Nov 2011 03:39:05 GMT
Content-MD5: p5/WA/oEr30qrEEl2lPAqw==
Authorization: AWS AKIAIOSFODNN7EXAMPLE:W0qPYICLE6JwkZAD1ei6hp9XZIee=
Content-Length: 79
Connection: Keep-Alive

<Delete>
  <Object>
    <Key>SampleDocument.txt</Key>
  </Object>
</Delete>
```

Sample Response

Because versioning is enabled on the bucket, Amazon S3 does not delete the object. Instead, it adds a delete marker for this object. The following response indicates that a delete marker was added (the DeleteMarker element in the response as a value of true) and the version number of the delete marker it added.

```xml
HTTP/1.1 200 OK
x-amz-id-2: P3xqrhuhYx1refdw3rEzmJh8z5KDtGzb+/FB7oiQaScI9Yaxd8olYXc7d111ab+
x-amz-request-id: 264A17BF16E9E80A
Date: Wed, 30 Nov 2011 03:39:32 GMT
Content-Type: application/xml
Server: AmazonS3
Content-Length: 276

<?xml version="1.0" encoding="UTF-8"?>
  <Deleted>
    <Key>SampleDocument.txt</Key>
    <DeleteMarker>true</DeleteMarker>
    <DeleteMarkerVersionId>NeQt5xeFTfgPJDB84CGWnkSLtluMr11s</DeleteMarkerVersionId>
  </Deleted>
</DeleteResult>
```

Case 2 - Versioned Delete

The following request attempts to delete a specific version of an object.

```xml
POST /delete HTTP/1.1
Host: bucketname.s3.<Region>.amazonaws.com
Accept: */*
x-amz-date: Wed, 30 Nov 2011 03:39:05 GMT
Content-MD5: p5/WA/oEr30qrEEl2lPAqw==
Authorization: AWS AKIAIOSFODNN7EXAMPLE:W0qPYICLE6JwkZAD1ei6hp9XZIee=
```
Sample Response

In this case, Amazon S3 deletes the specific object version from the bucket and returns the following response. In the response, Amazon S3 returns the key and version ID of the object deleted.

HTTP/1.1 200 OK
x-amz-id-2: P3xqrhuhyXlrefdw3EzmJh8z5KDtGzb+/FB7olQaSc19Yaxd801YXc7d1111xx
x-amz-request-id: 264A17BF16E9E80A
Date: Wed, 30 Nov 2011 03:39:32 GMT
Content-Type: application/xml
Server: AmazonS3
Content-Length: 219

<?xml version="1.0" encoding="UTF-8"?>
<DeleteResult xmlns="http://s3.amazonaws.com/doc/2006-03-01/">
  <Deleted>
    <Key>SampleDocument.txt</Key>
    <VersionId>OYcLXagmS.WaD..oyH4KRguB95_YhLs7</VersionId>
  </Deleted>
</DeleteResult>

Case 3 - Versioned delete of a delete marker

In the preceding example, the request refers to a delete marker (instead of an object), then Amazon S3 deletes the delete marker. The effect of this operation is to make your object reappear in your bucket. Amazon S3 returns a response that indicates the delete marker it deleted (DeleteMarker element with value true) and the version ID of the delete marker.

HTTP/1.1 200 OK
x-amz-id-2: IIPUZrtolxDIfxW8e9JLS2e6yWfTy3HqJt2iAe0ZE4XHa6NRVAjCp51zZaBr
x-amz-request-id: D6B284BEC9B05E4E
Date: Wed, 30 Nov 2011 03:43:25 GMT
Content-Type: application/xml
Server: AmazonS3
Content-Length: 331

<?xml version="1.0" encoding="UTF-8"?>
<DeleteResult xmlns="http://s3.amazonaws.com/doc/2006-03-01/">
  <Deleted>
    <Key>SampleDocument.txt</Key>
    <VersionId>OYcLXagmS.WaD..oyH4KRguB95_YhLs7</VersionId>
    <DeleteMarker>true</DeleteMarker>
    <DeleteMarkerVersionId>NeQt5xeFTfgPJD8B4CGWnkSLtluMr11s</DeleteMarkerVersionId>
  </Deleted>
</DeleteResult>
Sample Response

In general, when a multi-object Delete request results in Amazon S3 either adding a delete marker or removing a delete marker, the response returns the following elements.

```xml
<DeleteMarker>true</DeleteMarker>
<DeleteMarkerVersionId>NeQt5xeFTfgPJD8B4CGWnkSLtluMr11s</DeleteMarkerVersionId>
```

Sample Request: Malformed XML in the request

This example shows how Amazon S3 responds to a request that includes a malformed XML document. The following request sends a malformed XML document (missing the Delete end element).

```plaintext
POST /?delete HTTP/1.1
Host: bucketname.s3.<Region>.amazonaws.com
Accept: */*
x-amz-date: Wed, 30 Nov 2011 03:39:05 GMT
Content-MD5: p5/WA/oEr30qrEEl21PAqw==
Authorization: AWS AKIAIOSFODNN7EXAMPLE:W0qPYCLe6JwkZAD1ei6hp9XZIee=
Content-Length: 104
Connection: Keep-Alive

/Delete
/Object
<Key>404.txt</Key>
</Object>
/Object
<Key>a.txt</Key>
</Object>
```

Sample Response

The response returns the error messages that describe the error.

```xml
HTTP/1.1 200 OK
x-amz-id-2: P3xqrhuhYxlrefdw3rEzmJh8z5KDtGzb+/FB7oiQAaScI9Yaxd8olYXc7d1111ab+
x-amz-request-id: 264A17BF16E9E80A
Date: Wed, 30 Nov 2011 03:39:32 GMT
Content-Type: application/xml
Server: AmazonS3
Content-Length: 207

<?xml version="1.0" encoding="UTF-8"?>
<Error>
<Code>MalformedXML</Code>
<Message>The XML you provided was not well-formed or did not validate against our published schema</Message>
<RequestId>264A17BF16E9E80A</RequestId>
<HostId>P3xqrhuhYxlrefdw3rEzmJh8z5KDtGzb+/FB7oiQAaScI9Yaxd8olYXc7d1111ab+</HostId>
</Error>
```

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:
• AWS Command Line Interface
• AWS SDK for .NET
• AWS SDK for C++
• AWS SDK for Go
• AWS SDK for Java
• AWS SDK for JavaScript
• AWS SDK for PHP V3
• AWS SDK for Python
• AWS SDK for Ruby V3
DeleteObjectTagging

Service: Amazon Simple Storage Service

Removes the entire tag set from the specified object. For more information about managing object tags, see Object Tagging.

To use this operation, you must have permission to perform the s3:DeleteObjectTagging action.

To delete tags of a specific object version, add the versionId query parameter in the request. You will need permission for the s3:DeleteObjectVersionTagging action.

The following operations are related to DeleteBucketMetricsConfiguration:

- PutObjectTagging
- GetObjectTagging

Request Syntax

```plaintext
DELETE /{Key+}?tagging&versionId=VersionId HTTP/1.1
Host: Bucket.s3.amazonaws.com
x-amz-expected-bucket-owner: ExpectedBucketOwner
```

URI Request Parameters

The request uses the following URI parameters.

**Bucket (p. 82)**

The bucket name containing the objects from which to remove the tags.

When using this API with an access point, you must direct requests to the access point hostname. The access point hostname takes the form `AccessPointName-AccountId.s3-accesspoint.Region.amazonaws.com`. When using this operation with an access point through the AWS SDKs, you provide the access point ARN in place of the bucket name. For more information about access point ARNs, see Using Access Points in the Amazon Simple Storage Service Developer Guide.

When using this API with Amazon S3 on Outposts, you must direct requests to the S3 on Outposts hostname. The S3 on Outposts hostname takes the form `AccessPointName-AccountId.outpostId.s3-outposts.Region.amazonaws.com`. When using this operation using S3 on Outposts through the AWS SDKs, you provide the Outposts bucket ARN in place of the bucket name. For more information about S3 on Outposts ARNs, see Using S3 on Outposts in the Amazon Simple Storage Service Developer Guide.

Required: Yes

**Key (p. 82)**

Name of the object key.

Length Constraints: Minimum length of 1.

Required: Yes

**versionId (p. 82)**

The versionId of the object that the tag-set will be removed from.
x-amz-expected-bucket-owner (p. 82)

The account id of the expected bucket owner. If the bucket is owned by a different account, the request will fail with an HTTP 403 (Access Denied) error.

Request Body

The request does not have a request body.

Response Syntax

HTTP/1.1 204
x-amz-version-id: VersionId

Response Elements

If the action is successful, the service sends back an HTTP 204 response.

The response returns the following HTTP headers.

x-amz-version-id (p. 83)

The versionId of the object the tag-set was removed from.

Examples

Sample Request

The following DELETE request deletes the tag set from the specified object.

```
DELETE /exampleobject?tagging HTTP/1.1
Host: examplebucket.s3.<Region>.amazonaws.com
Date: Wed, 25 Nov 2016 12:00:00 GMT
Authorization: signatureValue
```

Sample Response

The following successful response shows Amazon S3 returning a 204 No Content response. The tag set for the object has been removed.

```
HTTP/1.1 204 No Content
Date: Wed, 25 Nov 2016 12:00:00 GMT
Connection: close
Server: AmazonS3
```

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
• AWS SDK for Go
• AWS SDK for Java
• AWS SDK for JavaScript
• AWS SDK for PHP V3
• AWS SDK for Python
• AWS SDK for Ruby V3
DeletePublicAccessBlock
Service: Amazon Simple Storage Service

Removes the PublicAccessBlock configuration for an Amazon S3 bucket. To use this operation, you must have the s3:PutBucketPublicAccessBlock permission. For more information about permissions, see Permissions Related to Bucket Subresource Operations and Managing Access Permissions to Your Amazon S3 Resources.

The following operations are related to DeletePublicAccessBlock:

- Using Amazon S3 Block Public Access
- GetPublicAccessBlock
- PutPublicAccessBlock
- GetBucketPolicyStatus

Request Syntax

```
DELETE /?publicAccessBlock HTTP/1.1
Host: Bucket.s3.amazonaws.com
x-amz-expected-bucket-owner: ExpectedBucketOwner
```

URI Request Parameters

The request uses the following URI parameters.

**Bucket (p. 85)**

The Amazon S3 bucket whose PublicAccessBlock configuration you want to delete.

Required: Yes

**x-amz-expected-bucket-owner (p. 85)**

The account id of the expected bucket owner. If the bucket is owned by a different account, the request will fail with an HTTP 403 (Access Denied) error.

Request Body

The request does not have a request body.

Response Syntax

```
HTTP/1.1 204
```

Response Elements

If the action is successful, the service sends back an HTTP 204 response with an empty HTTP body.

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
• AWS SDK for Go
• AWS SDK for Java
• AWS SDK for JavaScript
• AWS SDK for PHP V3
• AWS SDK for Python
• AWS SDK for Ruby V3
GetBucketAccelerateConfiguration
Service: Amazon Simple Storage Service

This implementation of the GET operation uses the accelerate subresource to return the Transfer Acceleration state of a bucket, which is either Enabled or Suspended. Amazon S3 Transfer Acceleration is a bucket-level feature that enables you to perform faster data transfers to and from Amazon S3.

To use this operation, you must have permission to perform the s3:GetAccelerateConfiguration action. The bucket owner has this permission by default. The bucket owner can grant this permission to others. For more information about permissions, see Permissions Related to Bucket Subresource Operations and Managing Access Permissions to your Amazon S3 Resources in the Amazon Simple Storage Service Developer Guide.

You set the Transfer Acceleration state of an existing bucket to Enabled or Suspended by using the PutBucketAccelerateConfiguration operation.

A GET accelerate request does not return a state value for a bucket that has no transfer acceleration state. A bucket has no Transfer Acceleration state if a state has never been set on the bucket.

For more information about transfer acceleration, see Transfer Acceleration in the Amazon Simple Storage Service Developer Guide.

Related Resources
- PutBucketAccelerateConfiguration

Request Syntax

```
GET /?accelerate HTTP/1.1
Host: Bucket.s3.amazonaws.com
x-amz-expected-bucket-owner: ExpectedBucketOwner
```

URI Request Parameters

The request uses the following URI parameters.

**Bucket (p. 87)**

The name of the bucket for which the accelerate configuration is retrieved.

Required: Yes

**x-amz-expected-bucket-owner (p. 87)**

The account id of the expected bucket owner. If the bucket is owned by a different account, the request will fail with an HTTP 403 (Access Denied) error.

Request Body

The request does not have a request body.

Response Syntax

```
HTTP/1.1 200
<?xml version="1.0" encoding="UTF-8"?>
<AccelerateConfiguration>
    <Status>string</Status>
</AccelerateConfiguration>
```
Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in XML format by the service.

**AccelerateConfiguration (p. 87)**

Root level tag for the AccelerateConfiguration parameters.

Required: Yes

**Status (p. 87)**

The accelerate configuration of the bucket.

Type: String

Valid Values: Enabled | Suspended

Examples

This implementation of the GET operation returns the following responses.

Example

If the transfer acceleration state is set to Enabled on a bucket, the response is as follows:

```xml
  <Status>Enabled</Status>
</AccelerateConfiguration>
```

Example

If the transfer acceleration state is set to Suspended on a bucket, the response is as follows:

```xml
  <Status>Suspended</Status>
</AccelerateConfiguration>
```

Example

If the transfer acceleration state on a bucket has never been set to Enabled or Suspended, the response is as follows:

```xml
```

Retrieve the transfer acceleration configuration for a bucket

The following example shows a GET /?accelerate request to retrieve the transfer acceleration state of the bucket named examplebucket.
  <Status>Enabled</Status>
</AccelerateConfiguration>

Example

The following is a sample of the response body (only) that shows bucket transfer acceleration is enabled.

GET /?accelerate HTTP/1.1
Host: examplebucket.s3.<Region>.amazonaws.com
Date: Mon, 11 Apr 2016 12:00:00 GMT
Authorization: authorization string
Content-Type: text/plain

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3
GetBucketAcl
Service: Amazon Simple Storage Service

This implementation of the GET operation uses the acl subresource to return the access control list (ACL) of a bucket. To use GET to return the ACL of the bucket, you must have READ_ACP access to the bucket. If READ_ACP permission is granted to the anonymous user, you can return the ACL of the bucket without using an authorization header.

Related Resources

- ListObjects

Request Syntax

GET /?acl HTTP/1.1
Host: Bucket.s3.amazonaws.com
x-amz-expected-bucket-owner: ExpectedBucketOwner

URI Request Parameters

The request uses the following URI parameters.

Bucket (p. 90)

Specifies the S3 bucket whose ACL is being requested.

Required: Yes

x-amz-expected-bucket-owner (p. 90)

The account id of the expected bucket owner. If the bucket is owned by a different account, the request will fail with an HTTP 403 (Access Denied) error.

Request Body

The request does not have a request body.

Response Syntax

HTTP/1.1 200
<?xml version="1.0" encoding="UTF-8"?>
<AccessControlPolicy>
  <Owner>
    <DisplayName>string</DisplayName>
    <ID>string</ID>
  </Owner>
  <AccessControlList>
    <Grant>
      <Grantee>
        <DisplayName>string</DisplayName>
        <EmailAddress>string</EmailAddress>
        <ID>string</ID>
        <xsi:type>string</xsi:type>
        <URI>string</URI>
      </Grantee>
      <Permission>string</Permission>
    </Grant>
  </AccessControlList>
</AccessControlPolicy>
Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in XML format by the service.

**AccessControlPolicy (p. 90)**

Root level tag for the AccessControlPolicy parameters.

Required: Yes

**Grants (p. 90)**

A list of grants.

Type: Array of Grant (p. 566) data types

**Owner (p. 90)**

Container for the bucket owner's display name and ID.

Type: Owner (p. 617) data type

Examples

Sample Request

The following request returns the ACL of the specified bucket.

```
GET /acl HTTP/1.1
Host: bucket.s3.<Region>.amazonaws.com
Date: Wed, 28 Oct 2009 22:32:00 GMT
Authorization: authorization string
```

Sample Response

```
HTTP/1.1 200 OK
x-amz-id-2: eftixk72aD6Ap51Tng0cOF8eF1dJ9G9L/2mkiDFu8yU9AS1ed4OpIszj7UDNEHGran
x-amz-request-id: 318BC6BC148832E5
Date: Wed, 28 Oct 2009 22:32:00 GMT
Last-Modified: Sun, 1 Jan 2006 12:00:00 GMT
Content-Length: 124
Content-Type: text/plain
Connection: close
Server: AmazonS3

<AccessControlPolicy>
  <Owner>
    <ID>75aa57f09aa0c8caeab4f8c24e99d10f8e7faeebf76c078efc7c6caea54ba06a</ID>
    <DisplayName>CustomersName@amazon.com</DisplayName>
  </Owner>
  <AccessControlList>
    <Grant>
      <Grantee xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:type="CanonicalUser">
        <ID>75aa57f09aa0c8caeab4f8c24e99d10f8e7faeebf76c078efc7c6caea54ba06a</ID>
        <DisplayName>CustomersName@amazon.com</DisplayName>
      </Grantee>
    </Grant>
  </AccessControlList>
</AccessControlPolicy>
```
See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3
GetBucketAnalyticsConfiguration
Service: Amazon Simple Storage Service

This implementation of the GET operation returns an analytics configuration (identified by the analytics configuration ID) from the bucket.

To use this operation, you must have permissions to perform the s3:GetAnalyticsConfiguration action. The bucket owner has this permission by default. The bucket owner can grant this permission to others. For more information about permissions, see Permissions Related to Bucket Subresource Operations and Managing Access Permissions to Your Amazon S3 Resources in the Amazon Simple Storage Service Developer Guide.

For information about Amazon S3 analytics feature, see Amazon S3 Analytics – Storage Class Analysis in the Amazon Simple Storage Service Developer Guide.

Related Resources
- DeleteBucketAnalyticsConfiguration
- ListBucketAnalyticsConfigurations
- PutBucketAnalyticsConfiguration

Request Syntax

GET /?analytics&id=Id HTTP/1.1
Host: Bucket.s3.amazonaws.com
x-amz-expected-bucket-owner: ExpectedBucketOwner

URI Request Parameters

The request uses the following URI parameters.

Bucket (p. 93)

The name of the bucket from which an analytics configuration is retrieved.

Required: Yes

id (p. 93)

The ID that identifies the analytics configuration.

Required: Yes

x-amz-expected-bucket-owner (p. 93)

The account id of the expected bucket owner. If the bucket is owned by a different account, the request will fail with an HTTP 403 (Access Denied) error.

Request Body

The request does not have a request body.

Response Syntax

HTTP/1.1 200
<?xml version="1.0" encoding="UTF-8"?>
Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in XML format by the service.

**AnalyticsConfiguration (p. 93)**

Root level tag for the AnalyticsConfiguration parameters.

   Required: Yes

**Filter (p. 93)**

The filter used to describe a set of objects for analyses. A filter must have exactly one prefix, one tag, or one conjunction (AnalyticsAndOperator). If no filter is provided, all objects will be considered in any analysis.

   Type: AnalyticsFilter (p. 521) data type

**Id (p. 93)**

The ID that identifies the analytics configuration.

   Type: String

**StorageClassAnalysis (p. 93)**

Contains data related to access patterns to be collected and made available to analyze the tradeoffs between different storage classes.

   Type: StorageClassAnalysis (p. 664) data type
Examples

Configure an Analytics Report

The following GET request for the bucket examplebucket returns the inventory configuration with the ID list1:

GET /?analytics&id=list1 HTTP/1.1
Host: examplebucket.s3.<Region>.amazonaws.com
Date: Mon, 31 Oct 2016 12:00:00 GMT
Authorization: authorization string

Example

The following is a sample response to the preceding GET request.

HTTP/1.1 200 OK
x-amz-id-2: YgIPIfBiKa2bj0KMgUAdQkf3ShJTOOpXUueF6QKo
x-amz-request-id: 236A8905248E5A02
Date: Mon, 31 Oct 2016 12:00:00 GMT
Server: AmazonS3
Content-Length: length

<?xml version="1.0" encoding="UTF-8"?>
<AnalyticsConfiguration xmlns="http://s3.amazonaws.com/doc/2006-03-01/"
   <Id>list1</Id>
   <Filter>
     <And>
       <Prefix>images/</Prefix>
       <Tag>
         <Key>dog</Key>
         <Value>corgi</Value>
       </Tag>
     </And>
   </Filter>
   <StorageClassAnalysis>
     <DataExport>
       <OutputSchemaVersion>V_1</OutputSchemaVersion>
       <Destination>
         <S3BucketDestination>
           <Format>CSV</Format>
           <BucketAccountId>123456789012</BucketAccountId>
           <Bucket>arn:aws:s3:::destination-bucket</Bucket>
           <Prefix>destination-prefix</Prefix>
         </S3BucketDestination>
       </Destination>
     </DataExport>
   </StorageClassAnalysis>
</AnalyticsConfiguration>

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
• AWS SDK for Go
• AWS SDK for Java
• AWS SDK for JavaScript
• AWS SDK for PHP V3
• AWS SDK for Python
• AWS SDK for Ruby V3
GetBucketCors
Service: Amazon Simple Storage Service

Returns the cors configuration information set for the bucket.

To use this operation, you must have permission to perform the s3:GetBucketCORS action. By default, the bucket owner has this permission and can grant it to others.

For more information about cors, see Enabling Cross-Origin Resource Sharing.

The following operations are related to GetBucketCors:

• PutBucketCors
• DeleteBucketCors

Request Syntax

GET /?cors HTTP/1.1
Host: Bucket.s3.amazonaws.com
x-amz-expected-bucket-owner: ExpectedBucketOwner

URI Request Parameters

The request uses the following URI parameters.

Bucket (p. 97)

The bucket name for which to get the cors configuration.

Required: Yes

x-amz-expected-bucket-owner (p. 97)

The account id of the expected bucket owner. If the bucket is owned by a different account, the request will fail with an HTTP 403 (Access Denied) error.

Request Body

The request does not have a request body.

Response Syntax

HTTP/1.1 200
<?xml version="1.0" encoding="UTF-8"?>
<CORSConfiguration>
  <CORSRule>
    <AllowedHeader>string</AllowedHeader>
    ...
    <AllowedMethod>string</AllowedMethod>
    ...
    <AllowedOrigin>string</AllowedOrigin>
    ...
    <ExposeHeader>string</ExposeHeader>
    ...
    <MaxAgeSeconds>integer</MaxAgeSeconds>
  </CORSRule>
  ...
</CORSConfiguration>
Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in XML format by the service.

**CORSConfiguration** (p. 97)

Root level tag for the CORSConfiguration parameters.

Required: Yes

**CORSRule** (p. 97)

A set of origins and methods (cross-origin access that you want to allow). You can add up to 100 rules to the configuration.

Type: Array of **CORSRule** (p. 536) data types

Examples

Configure cors Sample Request

The following PUT request adds the cors subresource to a bucket (examplebucket).

```
PUT /?cors HTTP/1.1
Host: examplebucket.s3.<Region>.amazonaws.com
x-amz-date: Tue, 21 Aug 2012 17:54:50 GMT
Content-MD5: 8dYiLewFWZyGvQpQF14W==
Authorization: authorization string
Content-Length: 216

<CORSConfiguration>
  <CORSRule>
    <AllowedOrigin>http://www.example.com</AllowedOrigin>
    <AllowedMethod>PUT</AllowedMethod>
    <AllowedMethod>POST</AllowedMethod>
    <AllowedMethod>DELETE</AllowedMethod>
    <AllowedHeader>*</AllowedHeader>
    <MaxAgeSeconds>3000</MaxAgeSec>
    <ExposeHeader>x-amz-server-side-encryption</ExposeHeader>
  </CORSRule>
  <CORSRule>
    <AllowedOrigin>*</AllowedOrigin>
    <AllowedMethod>GET</AllowedMethod>
    <AllowedHeader>*</AllowedHeader>
    <MaxAgeSeconds>3000</MaxAgeSeconds>
  </CORSRule>
</CORSConfiguration>
```

Example

This is the sample response to the preceding request.

```
HTTP/1.1 200 OK
x-amz-id-2: CCshOvbOPFxzhwOADyC4qHj/Ck3F9Q0viXXw3rivZ+GcBoZSO0ahvEjFpisZB7B
x-amz-request-id: BDC4B83DF5096BBB
Date: Tue, 21 Aug 2012 17:54:50 GMT
```

API Version 2006-03-01

98
Sample Request: Retrieve cors subresource

The following example gets the cors subresource of a bucket.

GET /?cors HTTP/1.1
Host: examplebucket.s3.<Region>.amazonaws.com
Date: Tue, 13 Dec 2011 19:14:42 GMT
Authorization: signatureValue

Sample Response

HTTP/1.1 200 OK
x-amz-id-2: 0FmFIWsh/PpBuzZ0JFRc552GVMqW45hJ7xVDqKwhEdJmf3q63RtrvH8ZuxW1Bo15
x-amz-request-id: 0CF038E9BCF63097
Date: Tue, 13 Dec 2011 19:14:42 GMT
Server: AmazonS3
Content-Length: 280
<CORSConfiguration>
  <CORSRule>
    <AllowedOrigin>http://www.example.com</AllowedOrigin>
    <AllowedMethod>GET</AllowedMethod>
    <MaxAgeSeconds>3000</MaxAgeSec>
    <ExposeHeader>x-amz-server-side-encryption</ExposeHeader>
  </CORSRule>
</CORSConfiguration>

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:
GetBucketEncryption
Service: Amazon Simple Storage Service

Returns the default encryption configuration for an Amazon S3 bucket. For information about the Amazon S3 default encryption feature, see Amazon S3 Default Bucket Encryption.

To use this operation, you must have permission to perform the s3:GetEncryptionConfiguration action. The bucket owner has this permission by default. The bucket owner can grant this permission to others. For more information about permissions, see Permissions Related to Bucket Subresource Operations and Managing Access Permissions to Your Amazon S3 Resources.

The following operations are related to GetBucketEncryption:

- PutBucketEncryption
- DeleteBucketEncryption

Request Syntax

GET /?encryption HTTP/1.1
Host: Bucket.s3.amazonaws.com
x-amz-expected-bucket-owner: ExpectedBucketOwner

URI Request Parameters

The request uses the following URI parameters.

Bucket (p. 100)

The name of the bucket from which the server-side encryption configuration is retrieved.

Required: Yes

x-amz-expected-bucket-owner (p. 100)

The account id of the expected bucket owner. If the bucket is owned by a different account, the request will fail with an HTTP 403 (Access Denied) error.

Request Body

The request does not have a request body.

Response Syntax

HTTP/1.1 200
<?xml version="1.0" encoding="UTF-8"?>
<ServerSideEncryptionConfiguration>
  <Rule>
    <ApplyServerSideEncryptionByDefault>
      <KMSMasterKeyID>string</KMSMasterKeyID>
      <SSEAlgorithm>string</SSEAlgorithm>
    </ApplyServerSideEncryptionByDefault>
  </Rule>
  ...
</ServerSideEncryptionConfiguration>

Response Elements

If the action is successful, the service sends back an HTTP 200 response.
The following data is returned in XML format by the service.

**ServerSideEncryptionConfiguration (p. 100)**

Root level tag for the ServerSideEncryptionConfiguration parameters.

Required: Yes

**Rule (p. 100)**

Container for information about a particular server-side encryption configuration rule.

Type: Array of **ServerSideEncryptionRule (p. 657)** data types

**Examples**

**Sample Request: Retrieve the encryption configuration for an S3 bucket**

The following example shows a GET /?encryption request.

```
GET /?encryption HTTP/1.1
Host: examplebucket.s3.<Region>.amazonaws.com
Date: Wed, 06 Sep 2017 12:00:00 GMT
Authorization: authorization string
Content-Length: length
```

**Sample Response**

This example illustrates one usage of GetBucketEncryption.

```
HTTP/1.1 200 OK
x-amz-id-2: kDmqsuw5Fدموxiوx9A4NJ/P1iE0c1rAU/ue2Yp60toXs4I5k5fqlwZsA6fV
x-amz-request-id: 5D8706FCB267387D
Date: Wed, 06 Sep 2017 12:00:00 GMT
Transfer-Encoding: chunked
Server: AmazonS3

  <Rule>
    <ApplyServerSideEncryptionByDefault>
      <SSEAlgorithm>aws:kms</SSEAlgorithm>
      <KMSMasterKeyID>arn:aws:kms:us-east-1:1234/5678example</KMSMasterKeyID>
    </ApplyServerSideEncryptionByDefault>
  </Rule>
</ServerSideEncryptionConfiguration>
```

**See Also**

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3
GetBucketIntelligentTieringConfiguration
Service: Amazon Simple Storage Service

Gets the S3 Intelligent-Tiering configuration from the specified bucket.

The S3 Intelligent-Tiering storage class is designed to optimize storage costs by automatically moving data to the most cost-effective storage access tier, without additional operational overhead. S3 Intelligent-Tiering delivers automatic cost savings by moving data between access tiers, when access patterns change.

The S3 Intelligent-Tiering storage class is suitable for objects larger than 128 KB that you plan to store for at least 30 days. If the size of an object is less than 128 KB, it is not eligible for auto-tiering. Smaller objects can be stored, but they are always charged at the frequent access tier rates in the S3 Intelligent-Tiering storage class.

If you delete an object before the end of the 30-day minimum storage duration period, you are charged for 30 days. For more information, see Storage class for automatically optimizing frequently and infrequently accessed objects.

Operations related to GetBucketIntelligentTieringConfiguration include:

- DeleteBucketIntelligentTieringConfiguration
- PutBucketIntelligentTieringConfiguration
- ListBucketIntelligentTieringConfigurations

Request Syntax

GET /?intelligent-tiering&id=Id HTTP/1.1
Host: Bucket.s3.amazonaws.com

URI Request Parameters

The request uses the following URI parameters.

Bucket (p. 103)

The name of the Amazon S3 bucket whose configuration you want to modify or retrieve.

Required: Yes

id (p. 103)

The ID used to identify the S3 Intelligent-Tiering configuration.

Required: Yes

Request Body

The request does not have a request body.

Response Syntax

HTTP/1.1 200
<?xml version="1.0" encoding="UTF-8"?>
<IntelligentTieringConfiguration>
  <Id>string</Id>
</IntelligentTieringConfiguration>
<Filter>
    <And>
        <Prefix>string</Prefix>
        <Tag>
            <Key>string</Key>
            <Value>string</Value>
        </Tag>
    </And>
    <Prefix>string</Prefix>
    <Tag>
        <Key>string</Key>
        <Value>string</Value>
    </Tag>
</Filter>

<Response Elements>
If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in XML format by the service.

**IntelligentTieringConfiguration (p. 103)**
- Root level tag for the IntelligentTieringConfiguration parameters.
  - Required: Yes

**Filter (p. 103)**
- Specifies a bucket filter. The configuration only includes objects that meet the filter's criteria.
  - Type: IntelligentTieringFilter (p. 574) data type

**Id (p. 103)**
- The ID used to identify the S3 Intelligent-Tiering configuration.
  - Type: String

**Status (p. 103)**
- Specifies the status of the configuration.
  - Type: String
  - Valid Values: Enabled | Disabled

**Tiering (p. 103)**
- Specifies the S3 Intelligent-Tiering storage class tier of the configuration.
  - Type: Array of Tiering (p. 669) data types

**See Also**
For more information about using this API in one of the language-specific AWS SDKs, see the following:
• AWS Command Line Interface
• AWS SDK for .NET
• AWS SDK for C++
• AWS SDK for Go
• AWS SDK for Java
• AWS SDK for JavaScript
• AWS SDK for PHP V3
• AWS SDK for Python
• AWS SDK for Ruby V3
GetBucketInventoryConfiguration
Service: Amazon Simple Storage Service

Returns an inventory configuration (identified by the inventory configuration ID) from the bucket.

To use this operation, you must have permissions to perform the s3:GetInventoryConfiguration action. The bucket owner has this permission by default and can grant this permission to others. For more information about permissions, see Permissions Related to Bucket Subresource Operations and Managing Access Permissions to Your Amazon S3 Resources.

For information about the Amazon S3 inventory feature, see Amazon S3 Inventory.

The following operations are related to GetBucketInventoryConfiguration:

- DeleteBucketInventoryConfiguration
- ListBucketInventoryConfigurations
- PutBucketInventoryConfiguration

Request Syntax

GET /?inventory&id=Id HTTP/1.1
Host: Bucket.s3.amazonaws.com
x-amz-expected-bucket-owner: ExpectedBucketOwner

URI Request Parameters

The request uses the following URI parameters.

Bucket (p. 106)

   The name of the bucket containing the inventory configuration to retrieve.

   Required: Yes

id (p. 106)

   The ID used to identify the inventory configuration.

   Required: Yes

x-amz-expected-bucket-owner (p. 106)

   The account id of the expected bucket owner. If the bucket is owned by a different account, the request will fail with an HTTP 403 (Access Denied) error.

Request Body

The request does not have a request body.

Response Syntax

HTTP/1.1 200
<?xml version="1.0" encoding="UTF-8"?>
<InventoryConfiguration>
  <Destination>
    <S3BucketDestination>
      <AccountId>string</AccountId>
    </S3BucketDestination>
  </Destination>
</InventoryConfiguration>
Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in XML format by the service.

**InventoryConfiguration (p. 106)**

Root level tag for the InventoryConfiguration parameters.

- Required: Yes

**Destination (p. 106)**

Contains information about where to publish the inventory results.

- Type: InventoryDestination (p. 577) data type

**Filter (p. 106)**

Specifies an inventory filter. The inventory only includes objects that meet the filter's criteria.

- Type: InventoryFilter (p. 579) data type

**Id (p. 106)**

The ID used to identify the inventory configuration.

- Type: String

**IncludedObjectVersions (p. 106)**

Object versions to include in the inventory list. If set to All, the list includes all the object versions, which adds the version-related fields VersionId, IsLatest, and DeleteMarker to the list. If set to Current, the list does not contain these version-related fields.

- Type: String

Valid Values: All | Current
IsEnabled (p. 106)

Specifies whether the inventory is enabled or disabled. If set to `true`, an inventory list is generated. If set to `false`, no inventory list is generated.

Type: Boolean

OptionalFields (p. 106)

Contains the optional fields that are included in the inventory results.

Type: Array of strings

Valid Values: Size | LastModifiedDate | StorageClass | ETag | IsMultipartUploaded | ReplicationStatus | EncryptionStatus | ObjectLockRetainUntilDate | ObjectLockMode | ObjectLockLegalHoldStatus | IntelligentTieringAccessTier

Schedule (p. 106)

Specifies the schedule for generating inventory results.

Type: `InventorySchedule` (p. 582) data type

Examples

Sample Request: Configure an inventory report

The following GET request for the bucket `examplebucket` returns the inventory configuration with the ID `list1`.

```
GET /?inventory&id=list1 HTTP/1.1
Host: examplebucket.s3.<Region>.amazonaws.com
Date: Mon, 31 Oct 2016 12:00:00 GMT
Authorization: authorization string
```

Sample Response

This example illustrates one usage of `GetBucketInventoryConfiguration`.

```
HTTP/1.1 200 OK
x-amz-id-2: YgIPIfBiKa2bj0KMyqf3ShJTOOpXUueF6QKo
x-amz-request-id: 236A8905248E5A02
Date: Mon, 31 Oct 2016 12:00:00 GMT
Server: AmazonS3
Content-Length: length

<?xml version="1.0" encoding="UTF-8"?>
<InventoryConfiguration xmlns="http://s3.amazonaws.com/doc/2006-03-01/"
    <Id>report1</Id>
    <IsEnabled>true</IsEnabled>
    <Destination>
        <S3BucketDestination>
            <Format>CSV</Format>
            <AccountId>123456789012</AccountId>
            <Bucket>arn:aws:s3:::destination-bucket</Bucket>
            <Prefix>prefix1</Prefix>
            <SSE-S3/>
        </S3BucketDestination>
    </Destination>
</InventoryConfiguration>
```
</Destination>
</Schedule>
  <Frequency>Daily</Frequency>
</Schedule>
</Filter>
<Prefix>myprefix/</Prefix>
</Filter>
<IncludedObjectVersions>All</IncludedObjectVersions>
<OptionalFields>
<Field>Size</Field>
<Field>LastModifiedDate</Field>
<Field>ETag</Field>
<Field>StorageClass</Field>
<Field>IsMultipartUploaded</Field>
<Field>ReplicationStatus</Field>
<Field>ObjectLockRetainUntilDate</Field>
<Field>ObjectLockMode</Field>
<Field>ObjectLockLegalHoldStatus</Field>
</OptionalFields>
</InventoryConfiguration>

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3
GetBucketLifecycle
Service: Amazon Simple Storage Service

Important
For an updated version of this API, see GetBucketLifecycleConfiguration. If you configured a bucket lifecycle using the filter element, you should see the updated version of this topic. This topic is provided for backward compatibility.

Returns the lifecycle configuration information set on the bucket. For information about lifecycle configuration, see Object Lifecycle Management.

To use this operation, you must have permission to perform the s3:GetLifecycleConfiguration action. The bucket owner has this permission by default. The bucket owner can grant this permission to others. For more information about permissions, see Permissions Related to Bucket Subresource Operations and Managing Access Permissions to Your Amazon S3 Resources.

GetBucketLifecycle has the following special error:

- Error code: NoSuchLifecycleConfiguration
  - Description: The lifecycle configuration does not exist.
  - HTTP Status Code: 404 Not Found
  - SOAP Fault Code Prefix: Client

The following operations are related to GetBucketLifecycle:

- GetBucketLifecycleConfiguration
- PutBucketLifecycle
- DeleteBucketLifecycle

Request Syntax

```
GET /?lifecycle HTTP/1.1
Host: Bucket.s3.amazonaws.com
x-amz-expected-bucket-owner: ExpectedBucketOwner
```

URI Request Parameters

The request uses the following URI parameters.

Bucket (p. 110)

The name of the bucket for which to get the lifecycle information.

Required: Yes

x-amz-expected-bucket-owner (p. 110)

The account id of the expected bucket owner. If the bucket is owned by a different account, the request will fail with an HTTP 403 (Access Denied) error.

Request Body

The request does not have a request body.

Response Syntax

```
HTTP/1.1 200
```
<?xml version="1.0" encoding="UTF-8"?>
<LifecycleConfiguration>
  <Rule>
    <AbortIncompleteMultipartUpload>
      <DaysAfterInitiation>integer</DaysAfterInitiation>
    </AbortIncompleteMultipartUpload>
    <Expiration>
      <Date>timestamp</Date>
      <Days>integer</Days>
      <ExpiredObjectDeleteMarker>boolean</ExpiredObjectDeleteMarker>
    </Expiration>
    <ID>string</ID>
    <NoncurrentVersionExpiration>
      <NoncurrentDays>integer</NoncurrentDays>
    </NoncurrentVersionExpiration>
    <NoncurrentVersionTransition>
      <NoncurrentDays>integer</NoncurrentDays>
      <StorageClass>string</StorageClass>
    </NoncurrentVersionTransition>
    <Prefix>string</Prefix>
    <Status>string</Status>
    <Transition>
      <Date>timestamp</Date>
      <Days>integer</Days>
      <StorageClass>string</StorageClass>
    </Transition>
  </Rule>
  ...
</LifecycleConfiguration>

Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in XML format by the service.

LifecycleConfiguration (p. 110)

Root level tag for the LifecycleConfiguration parameters.

Required: Yes

Rule (p. 110)

Container for a lifecycle rule.

Type: Array of Rule (p. 647) data types

Examples

Sample Request: Retrieve a lifecycle subresource

This example is a GET request to retrieve the lifecycle subresource from the specified bucket, and an example response with the returned lifecycle configuration.

```
GET /?lifecycle HTTP/1.1
Host: examplebucket.s3.<Region>.amazonaws.com
x-amz-date: Thu, 15 Nov 2012 00:17:21 GMT
Authorization: signatureValue
```
Sample Response

This example illustrates one usage of GetBucketLifecycle.

```
HTTP/1.1 200 OK
x-amz-id-2: ITnGT1y4RyTmXa3rPi4hklTXouTfohccUjo0iCPjz6FnfIutBj3M7fPlW02SEWp
x-amz-request-id: 51991C342C575321
Date: Thu, 15 Nov 2012 00:17:23 GMT
Server: AmazonS3
Content-Length: 358

<?xml version="1.0" encoding="UTF-8"?>
<LifecycleConfiguration xmlns="http://s3.amazonaws.com/doc/2006-03-01/">
  <Rule>
    <ID>Archive and then delete rule</ID>
    <Prefix>projectdocs/</Prefix>
    <Status>Enabled</Status>
    <Transition>
      <Days>30</Days>
      <StorageClass>STANDARD_IA</StorageClass>
    </Transition>
    <Transition>
      <Days>365</Days>
      <StorageClass>GLACIER</StorageClass>
    </Transition>
    <Expiration>
      <Days>3650</Days>
    </Expiration>
  </Rule>
</LifecycleConfiguration>
```

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3
GetBucketLifecycleConfiguration
Service: Amazon Simple Storage Service

**Note**
Bucket lifecycle configuration now supports specifying a lifecycle rule using an object key name prefix, one or more object tags, or a combination of both. Accordingly, this section describes the latest API. The response describes the new filter element that you can use to specify a filter to select a subset of objects to which the rule applies. If you are using a previous version of the lifecycle configuration, it still works. For the earlier API description, see GetBucketLifecycle.

Returns the lifecycle configuration information set on the bucket. For information about lifecycle configuration, see Object Lifecycle Management.

To use this operation, you must have permission to perform the s3:GetLifecycleConfiguration action. The bucket owner has this permission, by default. The bucket owner can grant this permission to others. For more information about permissions, see Permissions Related to Bucket Subresource Operations and Managing Access Permissions to Your Amazon S3 Resources.

GetBucketLifecycleConfiguration has the following special error:

- **Error code:** NoSuchLifecycleConfiguration
  - **Description:** The lifecycle configuration does not exist.
  - **HTTP Status Code:** 404 Not Found
  - **SOAP Fault Code Prefix:** Client

The following operations are related to GetBucketLifecycleConfiguration:

- GetBucketLifecycle
- PutBucketLifecycle
- DeleteBucketLifecycle

**Request Syntax**

```
GET /?lifecycle HTTP/1.1
Host: Bucket.s3.amazonaws.com
x-amz-expected-bucket-owner: ExpectedBucketOwner
```

**URI Request Parameters**
The request uses the following URI parameters.

**Bucket (p. 113)**

The name of the bucket for which to get the lifecycle information.

Required: Yes

**x-amz-expected-bucket-owner (p. 113)**

The account id of the expected bucket owner. If the bucket is owned by a different account, the request will fail with an HTTP 403 (Access Denied) error.

**Request Body**
The request does not have a request body.
Response Syntax

```xml
HTTP/1.1 200
<?xml version="1.0" encoding="UTF-8"?>
<LifecycleConfiguration>
  <Rule>
    <AbortIncompleteMultipartUpload>
      <DaysAfterInitiation(integer)</DaysAfterInitiation>
    </AbortIncompleteMultipartUpload>
    <Expiration>
      <Date>timestamp</Date>
      <Days>integer</Days>
      <ExpiredObjectDeleteMarker>boolean</ExpiredObjectDeleteMarker>
    </Expiration>
    <Filter>
      <And>
        <Prefix>string</Prefix>
        <Tag>
          <Key>string</Key>
          <Value>string</Value>
        </Tag>
        ...
      </And>
    </Filter>
    <ID>string</ID>
    <NoncurrentVersionExpiration>
      <NoncurrentDays>integer</NoncurrentDays>
    </NoncurrentVersionExpiration>
    <NoncurrentVersionTransition>
      <NoncurrentDays>integer</NoncurrentDays>
      <StorageClass>string</StorageClass>
    </NoncurrentVersionTransition>
    ...
    <Prefix>string</Prefix>
    <Status>string</Status>
    <Transition>
      <Date>timestamp</Date>
      <Days>integer</Days>
      <StorageClass>string</StorageClass>
    </Transition>
    ...
  </Rule>
  ...
</LifecycleConfiguration>
```

Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in XML format by the service.

**LifecycleConfiguration (p. 114)**

Root level tag for the LifecycleConfiguration parameters.

Required: Yes

**Rule (p. 114)**

Container for a lifecycle rule.
Type: Array of LifecycleRule (p. 589) data types

Examples

Sample Request

This example illustrates one usage of GetBucketLifecycleConfiguration.

```
GET /?lifecycle HTTP/1.1
Host: examplebucket.s3.<Region>.amazonaws.com
x-amz-date: Thu, 15 Nov 2012 00:17:21 GMT
Authorization: signatureValue
```

Sample Response

This example illustrates one usage of GetBucketLifecycleConfiguration.

```
HTTP/1.1 200 OK
x-amz-id-2: ITnGT1y4RyTmXa3rPi4hk1TXouTf0hccUjo0iCPjz6FnfIutBj3M7fPGLWO2SEWp
x-amz-request-id: 51991C342C575321
Date: Thu, 15 Nov 2012 00:17:23 GMT
Server: AmazonS3
Content-Length: 358

<?xml version="1.0" encoding="UTF-8"?>
<LifecycleConfiguration xmlns="http://s3.amazonaws.com/doc/2006-03-01/">
  <Rule>
    <ID>Archive and then delete rule</ID>
    <Prefix>projectdocs/</Prefix>
    <Status>Enabled</Status>
    <Transition>
      <Days>30</Days>
      <StorageClass>STANDARD_IA</StorageClass>
    </Transition>
    <Transition>
      <Days>365</Days>
      <StorageClass>GLACIER</StorageClass>
    </Transition>
    <Expiration>
      <Days>3650</Days>
    </Expiration>
  </Rule>
</LifecycleConfiguration>
```

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for JavaScript
• AWS SDK for PHP V3
• AWS SDK for Python
• AWS SDK for Ruby V3
GetBucketLocation
Service: Amazon Simple Storage Service

Returns the Region the bucket resides in. You set the bucket's Region using the LocationConstraint request parameter in a CreateBucket request. For more information, see CreateBucket.

To use this implementation of the operation, you must be the bucket owner.

The following operations are related to GetBucketLocation:

- GetObject
- CreateBucket

Request Syntax

GET /?location HTTP/1.1
Host: Bucket.s3.amazonaws.com
x-amz-expected-bucket-owner: ExpectedBucketOwner

URI Request Parameters

The request uses the following URI parameters.

**Bucket (p. 117)**

The name of the bucket for which to get the location.

Required: Yes

**x-amz-expected-bucket-owner (p. 117)**

The account id of the expected bucket owner. If the bucket is owned by a different account, the request will fail with an HTTP 403 (Access Denied) error.

Request Body

The request does not have a request body.

Response Syntax

HTTP/1.1 200
<?xml version="1.0" encoding="UTF-8"?>
<LocationConstraint>
  <LocationConstraint>string</LocationConstraint>
</LocationConstraint>

Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in XML format by the service.

**LocationConstraint (p. 117)**

Root level tag for the LocationConstraint parameters.

Required: Yes
LocationConstraint (p. 117)

Specifies the Region where the bucket resides. For a list of all the Amazon S3 supported location constraints by Region, see Regions and Endpoints. Buckets in Region us-east-1 have a LocationConstraint of null.

Type: String

Valid Values: af-south-1 | ap-east-1 | ap-northeast-1 | ap-northeast-2 | ap-northeast-3 | ap-south-1 | ap-southeast-1 | ap-southeast-2 | ca-central-1 | cn-north-1 | cn-northwest-1 | EU | eu-central-1 | eu-north-1 | eu-south-1 | eu-west-1 | eu-west-2 | eu-west-3 | me-south-1 | sa-east-1 | us-east-2 | us-gov-east-1 | us-gov-west-1 | us-west-1 | us-west-2

Examples

Sample Request

The following request returns the Region of the specified bucket.

```
GET /?location HTTP/1.1
Host: myBucket.s3.amazonaws.com
Date: Tue, 09 Oct 2007 20:26:04 +0000
Authorization: signatureValue
```

Sample Response

This example illustrates one usage of GetBucketLocation.

```
<?xml version="1.0" encoding="UTF-8"?>
<LocationConstraint xmlns="http://s3.amazonaws.com/doc/2006-03-01/">Europe</LocationConstraint>
```

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3
GetBucketLogging
Service: Amazon Simple Storage Service

Returns the logging status of a bucket and the permissions users have to view and modify that status. To use GET, you must be the bucket owner.

The following operations are related to GetBucketLogging:

- CreateBucket
- PutBucketLogging

Request Syntax

GET /?logging HTTP/1.1
Host: Bucket.s3.amazonaws.com
x-amz-expected-bucket-owner: ExpectedBucketOwner

URI Request Parameters

The request uses the following URI parameters.

Bucket (p. 119)

The bucket name for which to get the logging information.

Required: Yes

x-amz-expected-bucket-owner (p. 119)

The account id of the expected bucket owner. If the bucket is owned by a different account, the request will fail with an HTTP 403 (Access Denied) error.

Request Body

The request does not have a request body.

Response Syntax

HTTP/1.1 200
<?xml version="1.0" encoding="UTF-8"?>
<BucketLoggingStatus>
  <LoggingEnabled>
    <TargetBucket>string</TargetBucket>
    <TargetGrants>
      <Grant>
        <Grantee>
          <DisplayName>string</DisplayName>
          <EmailAddress>string</EmailAddress>
          <ID>string</ID>
          <xsi:type>string</xsi:type>
          <URI>string</URI>
        </Grantee>
        <Permission>string</Permission>
      </Grant>
    </TargetGrants>
    <TargetPrefix>string</TargetPrefix>
  </LoggingEnabled>
</BucketLoggingStatus>
Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in XML format by the service.

BucketLoggingStatus (p. 119)

Root level tag for the BucketLoggingStatus parameters.

Required: Yes

LoggingEnabled (p. 119)

Describes where logs are stored and the prefix that Amazon S3 assigns to all log object keys for a bucket. For more information, see PUT Bucket logging in the Amazon Simple Storage Service API Reference.

Type: LoggingEnabled (p. 593) data type

Examples

Sample Request

The following request returns the logging status for mybucket.

GET ?logging HTTP/1.1
Host: mybucket.s3.<Region>.amazonaws.com
Date: Wed, 25 Nov 2009 12:00:00 GMT
Authorization: authorization string

Sample Response: Showing an enabled logging status

This example illustrates one usage of GetBucketLogging.

HTTP/1.1 200 OK
Date: Wed, 25 Nov 2009 12:00:00 GMT
Connection: close
Server: AmazonS3

<?xml version="1.0" encoding="UTF-8"?>
<BucketLoggingStatus xmlns="http://doc.s3.amazonaws.com/2006-03-01">
  <LoggingEnabled>
    <TargetBucket>mybucketlogs</TargetBucket>
    <TargetPrefix>mybucket-access_log-/</TargetPrefix>
    <TargetGrants>
      <Grant>
        <Grantee xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:type="AmazonCustomerByEmail">
          <EmailAddress>user@company.com</EmailAddress>
        </Grantee>
        <Permission>READ</Permission>
      </Grant>
    </TargetGrants>
  </LoggingEnabled>
</BucketLoggingStatus>
Sample Response: Showing a disabled logging status

This example illustrates one usage of GetBucketLogging.

HTTP/1.1 200 OK
Date: Wed, 25 Nov 2009 12:00:00 GMT
Connection: close
Server: AmazonS3

<?xml version="1.0" encoding="UTF-8"?>

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3
GetBucketMetricsConfiguration
Service: Amazon Simple Storage Service

Gets a metrics configuration (specified by the metrics configuration ID) from the bucket. Note that this doesn't include the daily storage metrics.

To use this operation, you must have permissions to perform the s3:GetMetricsConfiguration action. The bucket owner has this permission by default. The bucket owner can grant this permission to others. For more information about permissions, see Permissions Related to Bucket Subresource Operations and Managing Access Permissions to Your Amazon S3 Resources.

For information about CloudWatch request metrics for Amazon S3, see Monitoring Metrics with Amazon CloudWatch.

The following operations are related to GetBucketMetricsConfiguration:

- PutBucketMetricsConfiguration
- DeleteBucketMetricsConfiguration
- ListBucketMetricsConfigurations
- Monitoring Metrics with Amazon CloudWatch

Request Syntax

GET /?metrics&id=Id HTTP/1.1
Host: Bucket.s3.amazonaws.com
x-amz-expected-bucket-owner: ExpectedBucketOwner

URI Request Parameters

The request uses the following URI parameters.

Bucket (p. 122)

The name of the bucket containing the metrics configuration to retrieve.

Required: Yes

id (p. 122)

The ID used to identify the metrics configuration.

Required: Yes

x-amz-expected-bucket-owner (p. 122)

The account id of the expected bucket owner. If the bucket is owned by a different account, the request will fail with an HTTP 403 (Access Denied) error.

Request Body

The request does not have a request body.

Response Syntax

HTTP/1.1 200
<?xml version="1.0" encoding="UTF-8"?>
<MetricsConfiguration>
Response Elements

If the action is successful, the service sends back an HTTP 200 response. The following data is returned in XML format by the service.

**MetricsConfiguration (p. 122)**

Root level tag for the MetricsConfiguration parameters.

- Required: Yes

**Filter (p. 122)**

Specifies a metrics configuration filter. The metrics configuration will only include objects that meet the filter's criteria. A filter must be a prefix, a tag, or a conjunction (MetricsAndOperator).

- Type: MetricsFilter (p. 598) data type

**Id (p. 122)**

The ID used to identify the metrics configuration.

- Type: String

Examples

First Sample Request

Retrieve a metrics configuration that filters metrics based on a specified prefix.

```
GET /?metrics&id=Documents HTTP/1.1
Host: examplebucket.s3.<Region>.amazonaws.com
x-amz-date: Thu, 15 Nov 2016 00:17:21 GMT
Authorization: signatureValue
```

First Sample Response

This example illustrates one usage of GetBucketMetricsConfiguration.
HTTP/1.1 200 OK
x-amz-id-2: ITnGT1y4REXAMPLEPi4hklTXouTf0hccUjo0iCPFXAMPLEutBj3M7fPGLW02SEWp
x-amz-request-id: 51991EXAMPLE5321
Date: Thu, 15 Nov 2016 00:17:22 GMT
Server: AmazonS3
Content-Length: 180

<?xml version="1.0" encoding="UTF-8"?>
  <Id>Documents</Id>
  <Filter>
    <Prefix>documents/</Prefix>
  </Filter>
</MetricsConfiguration>

Second Sample Request

Retrieve a metrics configuration that enables metrics for objects that start with a particular prefix and also have specific tags applied.

GET /?metrics&id=ImportantBlueDocuments HTTP/1.1
Host: examplebucket.s3.<Region>.amazonaws.com
x-amz-date: Thu, 15 Nov 2016 00:17:21 GMT
Authorization: signatureValue

Second Sample Response

This example illustrates one usage of GetBucketMetricsConfiguration.

HTTP/1.1 200 OK
x-amz-id-2: ITnGT1y4REXAMPLEPi4hklTXouTf0hccUjo0iCPFXAMPLEutBj3M7fPGLW02SEWp
x-amz-request-id: 51991EXAMPLE5321
Date: Thu, 15 Nov 2016 00:17:22 GMT
Server: AmazonS3
Content-Length: 480

<?xml version="1.0" encoding="UTF-8"?>
  <Id>ImportantBlueDocuments</Id>
  <Filter>
    <And>
      <Prefix>documents/</Prefix>
      <Tag>
        <Key>priority</Key>
        <Value>high</Value>
      </Tag>
      <Tag>
        <Key>class</Key>
        <Value>blue</Value>
      </Tag>
    </And>
  </Filter>
</MetricsConfiguration>

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:
• AWS Command Line Interface
• AWS SDK for .NET
• AWS SDK for C++
• AWS SDK for Go
• AWS SDK for Java
• AWS SDK for JavaScript
• AWS SDK for PHP V3
• AWS SDK for Python
• AWS SDK for Ruby V3
GetBucketNotification
Service: Amazon Simple Storage Service

No longer used, see GetBucketNotificationConfiguration.

Request Syntax

```
GET /?notification HTTP/1.1
Host: Bucket.s3.amazonaws.com
x-amz-expected-bucket-owner: ExpectedBucketOwner
```

URI Request Parameters

The request uses the following URI parameters.

**Bucket (p. 126)**

The name of the bucket for which to get the notification configuration.

Required: Yes

**x-amz-expected-bucket-owner (p. 126)**

The account id of the expected bucket owner. If the bucket is owned by a different account, the request will fail with an HTTP 403 (Access Denied) error.

Request Body

The request does not have a request body.

Response Syntax

```
HTTP/1.1 200
<?xml version="1.0" encoding="UTF-8"?>
<NotificationConfiguration>
  <TopicConfiguration>
    <Event>string</Event>
    <Event>string</Event>
    ...
    <Id>string</Id>
    <Topic>string</Topic>
  </TopicConfiguration>
  <QueueConfiguration>
    <Event>string</Event>
    <Event>string</Event>
    ...
    <Id>string</Id>
    <Queue>string</Queue>
  </QueueConfiguration>
  <CloudFunctionConfiguration>
    <CloudFunction>string</CloudFunction>
    <Event>string</Event>
    <Event>string</Event>
    ...
    <Id>string</Id>
    <InvocationRole>string</InvocationRole>
  </CloudFunctionConfiguration>
</NotificationConfiguration>
```

API Version 2006-03-01

126
Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in XML format by the service.

**NotificationConfiguration (p. 126)**

Root level tag for the NotificationConfiguration parameters.

Required: Yes

**CloudFunctionConfiguration (p. 126)**

Container for specifying the AWS Lambda notification configuration.

Type: CloudFunctionConfiguration (p. 526) data type

**QueueConfiguration (p. 126)**

This data type is deprecated. This data type specifies the configuration for publishing messages to an Amazon Simple Queue Service (Amazon SQS) queue when Amazon S3 detects specified events.

Type: QueueConfigurationDeprecated (p. 629) data type

**TopicConfiguration (p. 126)**

This data type is deprecated. A container for specifying the configuration for publication of messages to an Amazon Simple Notification Service (Amazon SNS) topic when Amazon S3 detects specified events.

Type: TopicConfigurationDeprecated (p. 672) data type

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3
GetBucketNotificationConfiguration
Service: Amazon Simple Storage Service

Returns the notification configuration of a bucket.

If notifications are not enabled on the bucket, the operation returns an empty NotificationConfiguration element.

By default, you must be the bucket owner to read the notification configuration of a bucket. However, the bucket owner can use a bucket policy to grant permission to other users to read this configuration with the s3:GetBucketNotification permission.

For more information about setting and reading the notification configuration on a bucket, see Setting Up Notification of Bucket Events. For more information about bucket policies, see Using Bucket Policies.

The following operation is related to GetBucketNotification:

- PutBucketNotification

Request Syntax

GET /?notification HTTP/1.1
Host: Bucket.s3.amazonaws.com
x-amz-expected-bucket-owner: ExpectedBucketOwner

URI Request Parameters

The request uses the following URI parameters.

Bucket (p. 128)

The name of the bucket for which to get the notification configuration.

Required: Yes

x-amz-expected-bucket-owner (p. 128)

The account id of the expected bucket owner. If the bucket is owned by a different account, the request will fail with an HTTP 403 (Access Denied) error.

Request Body

The request does not have a request body.

Response Syntax

HTTP/1.1 200
<?xml version="1.0" encoding="UTF-8"?>
<NotificationConfiguration>
  <TopicConfiguration>
    <Event>string</Event>
    ...
    <Filter>
      <S3Key>
        <FilterRule>
          <Name>string</Name>
          <Value>string</Value>
        </FilterRule>
      </S3Key>
    </Filter>
  </TopicConfiguration>
</NotificationConfiguration>
Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in XML format by the service.

**NotificationConfiguration (p. 128)**

Root level tag for the NotificationConfiguration parameters.

Required: Yes

**CloudFunctionConfiguration (p. 128)**

Describes the AWS Lambda functions to invoke and the events for which to invoke them.

Type: Array of LambdaFunctionConfiguration (p. 585) data types

**QueueConfiguration (p. 128)**

The Amazon Simple Queue Service queues to publish messages to and the events for which to publish messages.

Type: Array of QueueConfiguration (p. 627) data types
TopicConfiguration (p. 128)

The topic to which notifications are sent and the events for which notifications are generated.

Type: Array of TopicConfiguration (p. 670) data types

Examples

Sample Request

This request returns the notification configuration on the bucket quotes.s3.<Region>.amazonaws.com.

```
GET ?notification HTTP/1.1
Host: quotes.s3.<Region>.amazonaws.com
Date: Wed, 15 Oct 2014 16:59:03 GMT
Authorization: authorization string
```

Sample Response

This response returns that the notification configuration for the specified bucket.

```
HTTP/1.1 200 OK
x-amz-id-2: YgIPIfBiKa2bj0KMgUAdQkf3ShJT0OOpXUueF6QKo
x-amz-request-id: 236A8905248E5A02
Date: Wed, 15 Oct 2014 16:59:04 GMT
Server: AmazonS3
<?xml version="1.0" encoding="UTF-8"?>
<TopicConfiguration>
  <Id>YjVkM2Y0YmUtNGI3NC00ZjQyLWEwNGItNDIyYWUxY2I0N2M4</Id>
  <Topic>arn:aws:sns:us-east-1:account-id:s3notificationtopic2</Topic>
  <Event>s3:ReducedRedundancyLostObject</Event>
  <Event>s3:ObjectCreated:*</Event>
</TopicConfiguration>
</NotificationConfiguration>
```

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3
GetBucketOwnershipControls
Service: Amazon Simple Storage Service

Retrieves OwnershipControls for an Amazon S3 bucket. To use this operation, you must have the s3:GetBucketOwnershipControls permission. For more information about Amazon S3 permissions, see Specifying Permissions in a Policy.

For information about Amazon S3 Object Ownership, see Using Object Ownership.

The following operations are related to GetBucketOwnershipControls:

- PutBucketOwnershipControls (p. 314)
- DeleteBucketOwnershipControls (p. 59)

Request Syntax

GET /?ownershipControls HTTP/1.1
Host: Bucket.s3.amazonaws.com
x-amz-expected-bucket-owner: ExpectedBucketOwner

URI Request Parameters

The request uses the following URI parameters.

**Bucket (p. 131)**

The name of the Amazon S3 bucket whose OwnershipControls you want to retrieve.

Required: Yes

**x-amz-expected-bucket-owner (p. 131)**

The account id of the expected bucket owner. If the bucket is owned by a different account, the request will fail with an HTTP 403 (Access Denied) error.

Request Body

The request does not have a request body.

Response Syntax

HTTP/1.1 200
<?xml version="1.0" encoding="UTF-8"?>
<OwnershipControls>
  <Rule>
    <ObjectOwnership>String</ObjectOwnership>
  </Rule>
  ...
</OwnershipControls>

Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in XML format by the service.
**OwnershipControls (p. 131)**

Root level tag for the OwnershipControls parameters.

Required: Yes

**Rule (p. 131)**

The container element for an ownership control rule.

Type: Array of OwnershipControlsRule (p. 619) data types

**Examples**

**Sample GetBucketOwnership Controls Request**

This example illustrates one usage of GetBucketOwnershipControls.

```
GET /example-bucket?/ownershipControls HTTP/1.1
Host: examplebucket.s3.<Region>.amazonaws.com
Date: Thu, 18 Jun 2017 00:17:22 GMT
Authorization: signatureValue;
```

**Sample GetBucketOwnership Controls Response**

This example illustrates one usage of GetBucketOwnershipControls.

```
HTTP/1.1 200 OK
x-amz-id-2: Adphn7MaAHDEq9mh5JmcTN8mzyVX0JhIztSiQNaqTnxXXcYi4uiZbYdwWC3JXmh/
XXVUUQw04V8=
x-amz-request-id: 252631E05F84A415
Date: Thu, 18 Jun 2020 00:17:22 GMT
Server: AmazonS3
Content-Length: 194

<OwnershipControls xmlns="http://s3.amazonaws.com/doc/2006-03-01/">
  <Rule>
    <ObjectOwnership>BucketOwnerPreferred</ObjectOwnership>
  </Rule>
</OwnershipControls>
```

**See Also**

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python

API Version 2006-03-01
- AWS SDK for Ruby V3
GetBucketPolicy
Service: Amazon Simple Storage Service

Returns the policy of a specified bucket. If you are using an identity other than the root user of the AWS account that owns the bucket, the calling identity must have the GetBucketPolicy permissions on the specified bucket and belong to the bucket owner's account in order to use this operation.

If you don't have GetBucketPolicy permissions, Amazon S3 returns a 403 Access Denied error. If you have the correct permissions, but you're not using an identity that belongs to the bucket owner's account, Amazon S3 returns a 405 Method Not Allowed error.

Important
As a security precaution, the root user of the AWS account that owns a bucket can always use this operation, even if the policy explicitly denies the root user the ability to perform this action.

For more information about bucket policies, see Using Bucket Policies and User Policies.

The following operation is related to GetBucketPolicy:

- GetObject

Request Syntax

GET /?policy HTTP/1.1
Host: Bucket.s3.amazonaws.com
x-amz-expected-bucket-owner: ExpectedBucketOwner

URI Request Parameters

The request uses the following URI parameters.

Bucket (p. 134)

The bucket name for which to get the bucket policy.

Required: Yes

x-amz-expected-bucket-owner (p. 134)

The account id of the expected bucket owner. If the bucket is owned by a different account, the request will fail with an HTTP 403 (Access Denied) error.

Request Body

The request does not have a request body.

Response Syntax

HTTP/1.1 200

{ Policy in JSON format }

Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.
Examples

Sample Request

The following request returns the policy of the specified bucket.

GET ?policy HTTP/1.1
Host: bucket.s3.<Region>.amazonaws.com
Date: Wed, 28 Oct 2009 22:32:00 GMT
Authorization: authorization string

Sample Response

This example illustrates one usage of GetBucketPolicy.

HTTP/1.1 200 OK
x-amz-id-2: Uuag1LuByru9p04SAMPLEAtRPfTaOFg==
x-amz-request-id: 656c76696e67SAMPLE57374
Date: Tue, 04 Apr 2010 20:34:56 GMT
Connection: keep-alive
Server: AmazonS3

{
    "Version":"2008-10-17",
    "Id":"aaaa-bbbb-cccc-dddd",
    "Statement" : [
        {
            "Effect":"Deny",
            "Sid":"1",
            "Principal" : {
                "AWS":"[111122223333","444455556666"]
            },
            "Action" : ["s3:*"],
            "Resource" : "arn:aws:s3:::bucket/*"
        }
    ]
}

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3
GetBucketPolicyStatus
Service: Amazon Simple Storage Service

Retrieves the policy status for an Amazon S3 bucket, indicating whether the bucket is public. In order to use this operation, you must have the s3:GetBucketPolicyStatus permission. For more information about Amazon S3 permissions, see Specifying Permissions in a Policy.

For more information about when Amazon S3 considers a bucket public, see The Meaning of “Public”.

The following operations are related to GetBucketPolicyStatus:

- Using Amazon S3 Block Public Access
- GetPublicAccessBlock
- PutPublicAccessBlock
- DeletePublicAccessBlock

Request Syntax

GET /?policyStatus HTTP/1.1
Host: Bucket.s3.amazonaws.com
x-amz-expected-bucket-owner: ExpectedBucketOwner

URI Request Parameters

The request uses the following URI parameters.

Bucket (p. 137)

The name of the Amazon S3 bucket whose policy status you want to retrieve.

Required: Yes

x-amz-expected-bucket-owner (p. 137)

The account id of the expected bucket owner. If the bucket is owned by a different account, the request will fail with an HTTP 403 (Access Denied) error.

Request Body

The request does not have a request body.

Response Syntax

HTTP/1.1 200
<?xml version="1.0" encoding="UTF-8"?>
<PolicyStatus>
  <IsPublic>boolean</IsPublic>
</PolicyStatus>

Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in XML format by the service.
**PolicyStatus (p. 137)**

Root level tag for the PolicyStatus parameters.

Required: Yes

**IsPublic (p. 137)**

The policy status for this bucket. TRUE indicates that this bucket is public. FALSE indicates that the bucket is not public.

Type: Boolean

**Examples**

**Sample Request**

The following request gets a bucket policy status.

```
GET /<bucket-name>?policyStatus HTTP/1.1
Host: <bucket-name>.s3.<Region>.amazonaws.com
x-amz-date: <Thu, 15 Nov 2016 00:17:21 GMT>
Authorization: <signatureValue>
```

**Sample Response**

This example illustrates one usage of GetBucketPolicyStatus.

```
HTTP/1.1 200 OK
x-amz-id-2: ITnGT1y4REXAMPLEPi4hklTXouTfohccUjo0iCPEXAMPLEutBj3M7fPGlW02SEWp
x-amz-request-id: 51991EXAMPLE5321
Date: Thu, 15 Nov 2016 00:17:22 GMT
Server: AmazonS3
Content-Length: 0

<PolicyStatus>
  <IsPublic>TRUE</IsPublic>
</PolicyStatus>
```

**See Also**

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3
GetBucketReplication
Service: Amazon Simple Storage Service

Returns the replication configuration of a bucket.

**Note**
It can take a while to propagate the put or delete a replication configuration to all Amazon S3 systems. Therefore, a get request soon after put or delete can return a wrong result.

For information about replication configuration, see Replication in the *Amazon Simple Storage Service Developer Guide*.

This operation requires permissions for the s3:GetReplicationConfiguration action. For more information about permissions, see Using Bucket Policies and User Policies.

If you include the Filter element in a replication configuration, you must also include the DeleteMarkerReplication and Priority elements. The response also returns those elements.

For information about GetBucketReplication errors, see List of replication-related error codes.

The following operations are related to GetBucketReplication:

- PutBucketReplication
- DeleteBucketReplication

**Request Syntax**

GET /?replication HTTP/1.1
Host: Bucket.s3.amazonaws.com
x-amz-expected-bucket-owner: ExpectedBucketOwner

**URI Request Parameters**

The request uses the following URI parameters.

**Bucket (p. 139)**

The bucket name for which to get the replication information.

Required: Yes

**x-amz-expected-bucket-owner (p. 139)**

The account id of the expected bucket owner. If the bucket is owned by a different account, the request will fail with an HTTP 403 (Access Denied) error.

**Request Body**

The request does not have a request body.

**Response Syntax**

HTTP/1.1 200
<?xml version="1.0" encoding="UTF-8"?>
<ReplicationConfiguration>
  <Role>string</Role>
  <Rule>
    <DeleteMarkerReplication>
      <Status>string</Status>
    </DeleteMarkerReplication>
  </Rule>
</ReplicationConfiguration>
Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in XML format by the service.

ReplicationConfiguration (p. 139)

Root level tag for the ReplicationConfiguration parameters.
Required: Yes

**Role (p. 139)**

The Amazon Resource Name (ARN) of the AWS Identity and Access Management (IAM) role that Amazon S3 assumes when replicating objects. For more information, see How to Set Up Replication in the Amazon Simple Storage Service Developer Guide.

Type: String

**Rule (p. 139)**

A container for one or more replication rules. A replication configuration must have at least one rule and can contain a maximum of 1,000 rules.

Type: Array of ReplicationRule (p. 636) data types

**Examples**

**Sample Request: Retrieve replication configuration information**

The following GET request retrieves information about the replication configuration set for the examplebucket bucket:

```
GET /?replication HTTP/1.1
Host: examplebucket.s3.<Region>.amazonaws.com
Date: Tue, 10 Feb 2015 00:17:21 GMT
Authorization: authorization string
```

**Sample Response**

The following response shows that replication is enabled on the bucket. The empty prefix indicates that Amazon S3 will replicate all objects that are created in the examplebucket bucket. The Destination element identifies the target bucket where Amazon S3 creates the object replicas, and the storage class (STANDARD_IA) that Amazon S3 uses when creating replicas.

Amazon S3 assumes the specified IAM role to replicate objects on behalf of the bucket owner, which is the AWS account that created the bucket.

```
HTTP/1.1 200 OK
x-amz-id-2: ITnGT1y4RyTmXa3rPi4hklTXouTf0hccUjo0iCPjz6FnfItuBj3M7fPlW02SEwp
x-amz-request-id: 51991C342example
Date: Tue, 10 Feb 2015 00:17:23 GMT
Server: AmazonS3
Content-Length: contentlength

<?xml version="1.0" encoding="UTF-8"?>
<ReplicationConfiguration>
  <Role>arn:aws:iam::35667example:role/CrossRegionReplicationRoleForS3</Role>
  <Rule>
    <ID>rule1</ID>
    <Status>Enabled</Status>
    <Priority>1</Priority>
    <DeleteMarkerReplication>
      <Status>Disabled</Status>
    </DeleteMarkerReplication>
    <Filter>
      <And>
```

API Version 2006-03-01

141
<Prefix>TaxDocs</Prefix>
<Tag>
<Key>key1</Key>
<Value>value1</Value>
</Tag>
<Tag>
<Key>key1</Key>
<Value>value1</Value>
</Tag>
</And>
</Filter>
<Destination>
<Bucket>arn:aws:s3:::exampletargetbucket</Bucket>
</Destination>
</Rule>
</ReplicationConfiguration>

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3
GetBucketRequestPayment
Service: Amazon Simple Storage Service

Returns the request payment configuration of a bucket. To use this version of the operation, you must be the bucket owner. For more information, see Requester Pays Buckets.

The following operations are related to GetBucketRequestPayment:

- ListObjects

Request Syntax

```plaintext
GET /?requestPayment HTTP/1.1
Host: Bucket.s3.amazonaws.com
x-amz-expected-bucket-owner: ExpectedBucketOwner
```

URI Request Parameters

The request uses the following URI parameters.

**Bucket (p. 143)**

The name of the bucket for which to get the payment request configuration

Required: Yes

**x-amz-expected-bucket-owner (p. 143)**

The account id of the expected bucket owner. If the bucket is owned by a different account, the request will fail with an HTTP 403 (Access Denied) error.

Request Body

The request does not have a request body.

Response Syntax

```xml
HTTP/1.1 200
<?xml version="1.0" encoding="UTF-8"?>
<RequestPaymentConfiguration>
  <Payer>string</Payer>
</RequestPaymentConfiguration>
```

Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in XML format by the service.

**RequestPaymentConfiguration (p. 143)**

Root level tag for the RequestPaymentConfiguration parameters.

Required: Yes

**Payer (p. 143)**

Specifies who pays for the download and request fees.
Type: String

Valid Values: Requester | BucketOwner

Examples

Sample Request

The following request returns the payer for the bucket, `colorpictures`.

```
GET ?requestPayment HTTP/1.1
Host: colorpictures.s3.<Region>.amazonaws.com
Date: Wed, 01 Mar 2009 12:00:00 GMT
Authorization: authorization string
```

Sample Response

This response shows that the bucket is a Requester Pays bucket, meaning the person requesting a download from this bucket pays the transfer fees.

```
HTTP/1.1 200 OK
x-amz-id-2: YgIPlfBiKa2bJ0KMg95r/Ozo3emzU4dzsD4rcKCHQUAdQkf3ShJTOOpXUueF6QKo
x-amz-request-id: 236A8905248E5A01
Date: Wed, 01 Mar 2009 12:00:00 GMT
Content-Type: [type]
Content-Length: 0
Connection: close
Server: AmazonS3

<?xml version="1.0" encoding="UTF-8"?>
<RequestPaymentConfiguration xmlns="http://s3.amazonaws.com/doc/2006-03-01/">
  <Payer>Requester</Payer>
</RequestPaymentConfiguration>
```

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3
GetBucketTagging
Service: Amazon Simple Storage Service

Returns the tag set associated with the bucket.

To use this operation, you must have permission to perform the s3:GetBucketTagging action. By default, the bucket owner has this permission and can grant this permission to others.

GetBucketTagging has the following special error:

- Error code: NoSuchTagSetError
  - Description: There is no tag set associated with the bucket.

The following operations are related to GetBucketTagging:

- PutBucketTagging
- DeleteBucketTagging

Request Syntax

```plaintext
GET /?tagging HTTP/1.1
Host: Bucket.s3.amazonaws.com
x-amz-expected-bucket-owner: ExpectedBucketOwner
```

URI Request Parameters

The request uses the following URI parameters.

**Bucket (p. 145)**

  The name of the bucket for which to get the tagging information.

  Required: Yes

**x-amz-expected-bucket-owner (p. 145)**

  The account id of the expected bucket owner. If the bucket is owned by a different account, the request will fail with an HTTP 403 (Access Denied) error.

Request Body

The request does not have a request body.

Response Syntax

```xml
HTTP/1.1 200
<?xml version="1.0" encoding="UTF-8"?>
<Tagging>
  <TagSet>
    <Tag>
      <Key>string</Key>
      <Value>string</Value>
    </Tag>
  </TagSet>
</Tagging>
```
Response Elements

If the action is successful, the service sends back an HTTP 200 response. The following data is returned in XML format by the service.

Tagging (p. 145)

Root level tag for the Tagging parameters.
Required: Yes

TagSet (p. 145)

Contains the tag set.
Type: Array of Tag (p. 666) data types

Examples

Sample Request

The following request returns the tag set of the specified bucket.

```
GET ?tagging HTTP/1.1
Host: bucket.s3.<Region>.amazonaws.com
Date: Wed, 28 Oct 2009 22:32:00 GMT
Authorization: authorization string
```

Sample Response

Delete the metric configuration with a specified ID, which disables the CloudWatch metrics with the ExampleMetrics value for the FilterId dimension.

```
HTTP/1.1 200 OK
Date: Wed, 25 Nov 2009 12:00:00 GMT
Connection: close
Server: AmazonS3

<Tagging>
  <TagSet>
    <Tag>
      <Key>Project</Key>
      <Value>Project One</Value>
    </Tag>
    <Tag>
      <Key>User</Key>
      <Value>jsmith</Value>
    </Tag>
  </TagSet>
</Tagging>
```

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3
GetBucketVersioning
Service: Amazon Simple Storage Service

Returns the versioning state of a bucket.

To retrieve the versioning state of a bucket, you must be the bucket owner.

This implementation also returns the MFA Delete status of the versioning state. If the MFA Delete status is enabled, the bucket owner must use an authentication device to change the versioning state of the bucket.

The following operations are related to GetBucketVersioning:

- GetObject
- PutObject
- DeleteObject

Request Syntax

GET /?versioning HTTP/1.1
Host: Bucket.s3.amazonaws.com
x-amz-expected-bucket-owner: ExpectedBucketOwner

URI Request Parameters

The request uses the following URI parameters.

Bucket (p. 148)

The name of the bucket for which to get the versioning information.

Required: Yes

x-amz-expected-bucket-owner (p. 148)

The account id of the expected bucket owner. If the bucket is owned by a different account, the request will fail with an HTTP 403 (Access Denied) error.

Request Body

The request does not have a request body.

Response Syntax

HTTP/1.1 200
<?xml version="1.0" encoding="UTF-8"?>
<VersioningConfiguration>
  <Status>string</Status>
  <MfaDelete>string</MfaDelete>
</VersioningConfiguration>

Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in XML format by the service.
**VersioningConfiguration (p. 148)**

Root level tag for the VersioningConfiguration parameters.

Required: Yes

**MFADelete (p. 148)**

Specifies whether MFA delete is enabled in the bucket versioning configuration. This element is only returned if the bucket has been configured with MFA delete. If the bucket has never been so configured, this element is not returned.

Type: String

Valid Values: Enabled | Disabled

**Status (p. 148)**

The versioning state of the bucket.

Type: String

Valid Values: Enabled | Suspended

**Examples**

**Example**

This example returns the versioning state of myBucket.

GET /?versioning HTTP/1.1
Host: myBucket.s3.<Region>.amazonaws.com
Date: Wed, 12 Oct 2009 17:50:00 GMT
Authorization: authorization string
Content-Type: text/plain

**Example**

There are three versioning states:

If you enabled versioning on a bucket, the response is:

```xml
  <Status>Enabled</Status>
</VersioningConfiguration>
```

**Example**

If you suspended versioning on a bucket, the response is:

```xml
  <Status>Suspended</Status>
</VersioningConfiguration>
```
Example

If you never enabled (or suspended) versioning on a bucket, the response is:

```xml
```

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3
GetBucketWebsite
Service: Amazon Simple Storage Service

Returns the website configuration for a bucket. To host website on Amazon S3, you can configure a bucket as website by adding a website configuration. For more information about hosting websites, see Hosting Websites on Amazon S3.

This GET operation requires the S3:GetBucketWebsite permission. By default, only the bucket owner can read the bucket website configuration. However, bucket owners can allow other users to read the website configuration by writing a bucket policy granting them the S3:GetBucketWebsite permission.

The following operations are related to GetBucketWebsite:

- DeleteBucketWebsite
- PutBucketWebsite

Request Syntax

```
GET /?website HTTP/1.1
Host: Bucket.s3.amazonaws.com
x-amz-expected-bucket-owner: ExpectedBucketOwner
```

URI Request Parameters

The request uses the following URI parameters.

**Bucket (p. 151)**

The bucket name for which to get the website configuration.

Required: Yes

**x-amz-expected-bucket-owner (p. 151)**

The account id of the expected bucket owner. If the bucket is owned by a different account, the request will fail with an HTTP 403 (Access Denied) error.

Request Body

The request does not have a request body.

Response Syntax

```
HTTP/1.1 200
<?xml version="1.0" encoding="UTF-8"?>
<WebsiteConfiguration>
  <RedirectAllRequestsTo>
    <HostName>string</HostName>
    <Protocol>string</Protocol>
  </RedirectAllRequestsTo>
  <IndexDocument>
    <Suffix>string</Suffix>
  </IndexDocument>
  <ErrorDocument>
    <Key>string</Key>
  </ErrorDocument>
  <RoutingRules>
    <RoutingRule>
```

API Version 2006-03-01

151
Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in XML format by the service.

WebsiteConfiguration (p. 151)

Root level tag for the WebsiteConfiguration parameters.

Required: Yes

ErrorDocument (p. 151)

The object key name of the website error document to use for 4XX class errors.

Type: ErrorDocument (p. 562) data type

IndexDocument (p. 151)

The name of the index document for the website (for example index.html).

Type: IndexDocument (p. 569) data type

RedirectAllRequestsTo (p. 151)

Specifies the redirect behavior of all requests to a website endpoint of an Amazon S3 bucket.

Type: RedirectAllRequestsTo (p. 634) data type

RoutingRules (p. 151)

Rules that define when a redirect is applied and the redirect behavior.

Type: Array of RoutingRule (p. 646) data types

Examples

Sample Request

This request retrieves website configuration on the specified bucket.

```
GET ?website HTTP/1.1
Host: example-bucket.s3.<Region>.amazonaws.com
Date: Thu, 27 Jan 2011 00:49:20 GMT
Authorization: AWS AKIAIOSFODNN7EXAMPLE:n0Nhek72Ufg/u7Sm5C1dqRLs8XX=
```
Sample Response

This example illustrates one usage of GetBucketWebsite.

```
HTTP/1.1 200 OK
x-amz-id-2: YgIPIfBiKa2bj0KMgUAdQkf3ShJTOLPpXUueF6QKo
x-amz-request-id: 3848CD259D811111
Date: Thu, 27 Jan 2011 00:49:26 GMT
Content-Length: 240
Content-Type: application/xml
Transfer-Encoding: chunked
Server: AmazonS3

<?xml version='1.0' encoding='UTF-8'?>
<WebsiteConfiguration xmlns="http://s3.amazonaws.com/doc/2006-03-01/">
  <IndexDocument>
    <Suffix>index.html</Suffix>
  </IndexDocument>
  <ErrorDocument>
    <Key>404.html</Key>
  </ErrorDocument>
</WebsiteConfiguration>
```

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3
GetObject
Service: Amazon Simple Storage Service

Retrieves objects from Amazon S3. To use GET, you must have READ access to the object. If you grant READ access to the anonymous user, you can return the object without using an authorization header.

An Amazon S3 bucket has no directory hierarchy such as you would find in a typical computer file system. You can, however, create a logical hierarchy by using object key names that imply a folder structure. For example, instead of naming an object sample.jpg, you can name it photos/2006/February/sample.jpg.

To get an object from such a logical hierarchy, specify the full key name for the object in the GET operation. For a virtual hosted-style request example, if you have the object photos/2006/February/sample.jpg, specify the resource as /photos/2006/February/sample.jpg. For a path-style request example, if you have the object photos/2006/February/sample.jpg in the bucket named examplebucket, specify the resource as /examplebucket/photos/2006/February/sample.jpg. For more information about request types, see HTTP Host Header Bucket Specification.

To distribute large files to many people, you can save bandwidth costs by using BitTorrent. For more information, see Amazon S3 Torrent. For more information about returning the ACL of an object, see GetObjectAcl.

If the object you are retrieving is stored in the S3 Glacier, S3 Glacier Deep Archive, S3 Intelligent-Tiering Archive, or S3 Intelligent-Tiering Deep Archive storage classes, before you can retrieve the object you must first restore a copy using RestoreObject. Otherwise, this operation returns an InvalidObjectStateError error. For information about restoring archived objects, see Restoring Archived Objects.

Encryption request headers, like x-amz-server-side-encryption, should not be sent for GET requests if your object uses server-side encryption with CMKs stored in AWS KMS (SSE-KMS) or server-side encryption with Amazon S3–managed encryption keys (SSE-S3). If your object does use these types of keys, you'll get an HTTP 400 BadRequest error.

If you encrypt an object by using server-side encryption with customer-provided encryption keys (SSE-C) when you store the object in Amazon S3, then when you GET the object, you must use the following headers:

- x-amz-server-side-encryption-customer-algorithm
- x-amz-server-side-encryption-customer-key
- x-amz-server-side-encryption-customer-key-MD5

For more information about SSE-C, see Server-Side Encryption (Using Customer-Provided Encryption Keys).

Assuming you have permission to read object tags (permission for the s3:GetObjectVersionTagging action), the response also returns the x-amz-tagging-count header that provides the count of number of tags associated with the object. You can use GetObjectTagging to retrieve the tag set associated with an object.

Permissions

You need the s3:GetObject permission for this operation. For more information, see Specifying Permissions in a Policy. If the object you request does not exist, the error Amazon S3 returns depends on whether you also have the s3:ListBucket permission.

- If you have the s3:ListBucket permission on the bucket, Amazon S3 will return an HTTP status code 404 ("no such key") error.
- If you don’t have the s3:ListBucket permission, Amazon S3 will return an HTTP status code 403 ("access denied") error.
Versioning

By default, the GET operation returns the current version of an object. To return a different version, use the versionId subresource.

**Note**
If the current version of the object is a delete marker, Amazon S3 behaves as if the object was deleted and includes x-amz-delete-marker: true in the response.

For more information about versioning, see PutBucketVersioning.

Overriding Response Header Values

There are times when you want to override certain response header values in a GET response. For example, you might override the Content-Disposition response header value in your GET request.

You can override values for a set of response headers using the following query parameters. These response header values are sent only on a successful request, that is, when status code 200 OK is returned. The set of headers you can override using these parameters is a subset of the headers that Amazon S3 accepts when you create an object. The response headers that you can override for the GET response are Content-Type, Content-Language, Expires, Cache-Control, Content-Disposition, and Content-Encoding. To override these header values in the GET response, you use the following request parameters.

**Note**
You must sign the request, either using an Authorization header or a presigned URL, when using these parameters. They cannot be used with an unsigned (anonymous) request.

- response-content-type
- response-content-language
- response-expires
- response-cache-control
- response-content-disposition
- response-content-encoding

Additional Considerations about Request Headers

If both of the If-Match and If-Unmodified-Since headers are present in the request as follows: If-Match condition evaluates to true, and; If-Unmodified-Since condition evaluates to false; then, S3 returns 200 OK and the data requested.

If both of the If-None-Match and If-Modified-Since headers are present in the request as follows: If-None-Match condition evaluates to false, and; If-Modified-Since condition evaluates to true; then, S3 returns 304 Not Modified response code.

For more information about conditional requests, see RFC 7232.

The following operations are related to GetObject:

- ListBuckets
- GetObjectAcl

Request Syntax

```
GET /Key+?partNumber=PartNumber&response-cache-control=ResponseCacheControl&response-content-disposition=ResponseContentDisposition&response-
```
URI Request Parameters

The request uses the following URI parameters.

**Bucket (p. 155)**

The bucket name containing the object.

When using this API with an access point, you must direct requests to the access point hostname. The access point hostname takes the form `AccessPointName-AccountId.s3-accesspoint.Region.amazonaws.com`. When using this operation with an access point through the AWS SDKs, you provide the access point ARN in place of the bucket name. For more information about access point ARNs, see Using Access Points in the Amazon Simple Storage Service Developer Guide.

When using this API with Amazon S3 on Outposts, you must direct requests to the S3 on Outposts hostname. The S3 on Outposts hostname takes the form `AccessPointName-AccountId.outpostId.s3-outposts.Region.amazonaws.com`. When using this operation using S3 on Outposts through the AWS SDKs, you provide the Outposts bucket ARN in place of the bucket name. For more information about S3 on Outposts ARNs, see Using S3 on Outposts in the Amazon Simple Storage Service Developer Guide.

**Required:** Yes

**If-Match (p. 155)**

Return the object only if its entity tag (ETag) is the same as the one specified, otherwise return a 412 (precondition failed).

**If-Modified-Since (p. 155)**

Return the object only if it has been modified since the specified time, otherwise return a 304 (not modified).

**If-None-Match (p. 155)**

Return the object only if its entity tag (ETag) is different from the one specified, otherwise return a 304 (not modified).

**If-Unmodified-Since (p. 155)**

Return the object only if it has not been modified since the specified time, otherwise return a 412 (precondition failed).

**Key (p. 155)**

Key of the object to get.

**Length Constraints:** Minimum length of 1.
Required: Yes

**partNumber (p. 155)**

Part number of the object being read. This is a positive integer between 1 and 10,000. Effectively performs a 'ranged' GET request for the part specified. Useful for downloading just a part of an object.

**Range (p. 155)**

Downloads the specified range bytes of an object. For more information about the HTTP Range header, see https://www.w3.org/Protocols/rfc2616/rfc2616-sec14.html#sec14.35.

**Note**

Amazon S3 doesn't support retrieving multiple ranges of data per GET request.

**response-cache-control (p. 155)**

Sets the Cache-Control header of the response.

**response-content-disposition (p. 155)**

Sets the Content-Disposition header of the response.

**response-content-encoding (p. 155)**

Sets the Content-Encoding header of the response.

**response-content-language (p. 155)**

Sets the Content-Language header of the response.

**response-content-type (p. 155)**

Sets the Content-Type header of the response.

**response-expires (p. 155)**

Sets the Expires header of the response.

**versionId (p. 155)**

VersionId used to reference a specific version of the object.

**x-amz-expected-bucket-owner (p. 155)**

The account id of the expected bucket owner. If the bucket is owned by a different account, the request will fail with an HTTP 403 (Access Denied) error.

**x-amz-request-payer (p. 155)**

Confirms that the requester knows that they will be charged for the request. Bucket owners need not specify this parameter in their requests. For information about downloading objects from requester pays buckets, see Downloading Objects in Requestor Pays Buckets in the Amazon S3 Developer Guide.

Valid Values: requester

**x-amz-server-side-encryption-customer-algorithm (p. 155)**

Specifies the algorithm to use to when encrypting the object (for example, AES256).

**x-amz-server-side-encryption-customer-key (p. 155)**

Specifies the customer-provided encryption key for Amazon S3 to use in encrypting data. This value is used to store the object and then it is discarded; Amazon S3 does not store the encryption key. The key must be appropriate for use with the algorithm specified in the x-amz-server-side-encryption-customer-algorithm header.
**x-amz-server-side-encryption-customer-key-MD5 (p. 155)**

Specifies the 128-bit MD5 digest of the encryption key according to RFC 1321. Amazon S3 uses this header for a message integrity check to ensure that the encryption key was transmitted without error.

**Request Body**

The request does not have a request body.

**Response Syntax**

```
HTTP/1.1 200
x-amz-delete-marker: DeleteMarker
accept-ranges: AcceptRanges
x-amz-expiration: Expiration
x-amz-restore: Restore
Last-Modified: LastModified
Content-Length: ContentLength
ETag: ETag
x-amz-missing-meta: MissingMeta
x-amz-version-id: VersionId
Cache-Control: CacheControl
Content-Disposition: ContentDisposition
Content-Encoding: ContentEncoding
Content-Language: ContentLanguage
Content-Range: ContentRange
Content-Type: ContentType
Expires: Expires
x-amz-website-redirect-location: WebsiteRedirectLocation
x-amz-server-side-encryption: ServerSideEncryption
x-amz-server-side-encryption-customer-algorithm: SSECustomerAlgorithm
x-amz-server-side-encryption-customer-key-MD5: SSECustomerKeyMD5
x-amz-server-side-encryption-aws-kms-key-id: SSEKMSKeyId
x-amz-storage-class: StorageClass
x-amz-request-charged: RequestCharged
x-amz-replication-status: ReplicationStatus
x-amz-mp-parts-count: PartsCount
x-amz-tagging-count: TagCount
x-amz-object-lock-mode: ObjectLockMode
x-amz-object-lock-retain-until-date: ObjectLockRetainUntilDate
x-amz-object-lock-legal-hold: ObjectLockLegalHoldStatus
```

**Response Elements**

If the action is successful, the service sends back an HTTP 200 response. The response returns the following HTTP headers.

**accept-ranges (p. 158)**

Indicates that a range of bytes was specified.

**Cache-Control (p. 158)**

Specifies caching behavior along the request/reply chain.

**Content-Disposition (p. 158)**

Specifies presentational information for the object.
Content-Encoding (p. 158)

Specifies what content encodings have been applied to the object and thus what decoding mechanisms must be applied to obtain the media-type referenced by the Content-Type header field.

Content-Language (p. 158)

The language the content is in.

Content-Length (p. 158)

Size of the body in bytes.

Content-Range (p. 158)

The portion of the object returned in the response.

Content-Type (p. 158)

A standard MIME type describing the format of the object data.

ETag (p. 158)

An ETag is an opaque identifier assigned by a web server to a specific version of a resource found at a URL.

Expires (p. 158)

The date and time at which the object is no longer cacheable.

Last-Modified (p. 158)

Last modified date of the object.

x-amz-delete-marker (p. 158)

Specifies whether the object retrieved was (true) or was not (false) a Delete Marker. If false, this response header does not appear in the response.

x-amz-expiration (p. 158)

If the object expiration is configured (see PUT Bucket lifecycle), the response includes this header. It includes the expiry-date and rule-id key-value pairs providing object expiration information. The value of the rule-id is URL encoded.

x-amz-missing-meta (p. 158)

This is set to the number of metadata entries not returned in x-amz-meta headers. This can happen if you create metadata using an API like SOAP that supports more flexible metadata than the REST API. For example, using SOAP, you can create metadata whose values are not legal HTTP headers.

x-amz-mp-parts-count (p. 158)

The count of parts this object has.

x-amz-object-lock-legal-hold (p. 158)

Indicates whether this object has an active legal hold. This field is only returned if you have permission to view an object's legal hold status.

Valid Values: ON | OFF

x-amz-object-lock-mode (p. 158)

The Object Lock mode currently in place for this object.

Valid Values: GOVERNANCE | COMPLIANCE

x-amz-object-lock-retain-until-date (p. 158)

The date and time when this object's Object Lock will expire.
Amazon Simple Storage Service API Reference
Amazon Simple Storage Service

x-amz-replication-status (p. 158)

Amazon S3 can return this if your request involves a bucket that is either a source or destination in a replication rule.

Valid Values: COMPLETE | PENDING | FAILED | REPLICA

x-amz-request-charged (p. 158)

If present, indicates that the requester was successfully charged for the request.

Valid Values: requester

x-amz-restore (p. 158)

Provides information about object restoration operation and expiration time of the restored object copy.

x-amz-server-side-encryption (p. 158)

The server-side encryption algorithm used when storing this object in Amazon S3 (for example, AES256, aws:kms).

Valid Values: AES256 | aws:kms

x-amz-server-side-encryption-aws-kms-key-id (p. 158)

If present, specifies the ID of the AWS Key Management Service (AWS KMS) symmetric customer managed customer master key (CMK) that was used for the object.

x-amz-server-side-encryption-customer-algorithm (p. 158)

If server-side encryption with a customer-provided encryption key was requested, the response will include this header confirming the encryption algorithm used.

x-amz-server-side-encryption-customer-key-MD5 (p. 158)

If server-side encryption with a customer-provided encryption key was requested, the response will include this header to provide round-trip message integrity verification of the customer-provided encryption key.

x-amz-storage-class (p. 158)

Provides storage class information of the object. Amazon S3 returns this header for all objects except for S3 Standard storage class objects.

Valid Values: STANDARD | REDUCED_REDUNDANCY | STANDARD_IA | ONEZONE_IA | INTELLIGENT_TIERING | GLACIER | DEEP_ARCHIVE | OUTPOSTS

x-amz-tagging-count (p. 158)

The number of tags, if any, on the object.

x-amz-version-id (p. 158)

Version of the object.

x-amz-website-redirect-location (p. 158)

If the bucket is configured as a website, redirects requests for this object to another object in the same bucket or to an external URL. Amazon S3 stores the value of this header in the object metadata.

The following data is returned in binary format by the service.

<varlistentry>  Body (p. 158)  </varlistentry>
The specified key does not exist. Object is archived and inaccessible until restored.

**Examples**

**Sample Request**

The following request returns the object `my-image.jpg`.

```plaintext
GET /my-image.jpg HTTP/1.1
Host: bucket.s3.<Region>.amazonaws.com
Date: Mon, 3 Oct 2016 22:32:00 GMT
Authorization: authorization string
```

**Sample Response**

This example illustrates one usage of GetObject.

```plaintext
HTTP/1.1 200 OK
x-amz-id-2: eftixk72aD6Ap51TngcoF8eFidJG9Z/2mkiDFu8yU9AS1ed4OpIszj7UDNEHGran
x-amz-request-id: 318BC8BC148832E5
Date: Mon, 3 Oct 2016 22:32:00 GMT
Last-Modified: Wed, 12 Oct 2009 17:50:00 GMT
ETag: "fba9dede5f27731c9771645a39863328"
Content-Length: 434234

[434234 bytes of object data]
```

**Sample Response: Object with associated tags**

If the object had tags associated with it, Amazon S3 returns the `x-amz-tagging-count` header with tag count.

```plaintext
HTTP/1.1 200 OK
x-amz-id-2: eftixk72aD6Ap51TngcoF8eFidJG9Z/2mkiDFu8yU9AS1ed4OpIszj7UDNEHGran
x-amz-request-id: 318BC8BC148832E5
Date: Mon, 3 Oct 2016 22:32:00 GMT
Last-Modified: Wed, 12 Oct 2009 17:50:00 GMT
ETag: "fba9dede5f27731c9771645a39863328"
Content-Length: 434234
x-amz-tagging-count: 2

[434234 bytes of object data]
```

**Sample Response: Object with an expiration**

If the object had expiration set using lifecycle configuration, you get the following response with the `x-amz-expiration` header.

```plaintext
HTTP/1.1 200 OK
x-amz-id-2: eftixk72aD6Ap51TngcoF8eFidJG9Z/2mkiDFu8yU9AS1ed4OpIszj7UDNEHGran
x-amz-request-id: 318BC8BC148832E5
Date: Wed, 28 Oct 2009 22:32:00 GMT
```
Sample Response if an Object Is Archived in Glacier

If the object you are retrieving is stored in the S3 Glacier, S3 Glacier Deep Archive, S3 Intelligent-Tiering Archive, or S3 Intelligent-Tiering Deep Archive storage classes, before you can retrieve the object you must first restore a copy using RestoreObject. Otherwise, this operation returns an InvalidObjectStateError error. If restoring from Archive Access tier or Deep Archive Access tier the response will additionally include StorageClass and AccessTier elements.

HTTP/1.1 403 Forbidden
x-amz-request-id: CD4BD8A1310A11B3
x-amz-id-2: m9RDbQU0+RRB7jmUN1ChqLeqMUnr9dv8b+KP612gHfRJZSTsrMCoRP8RtFrzX9mb
Content-Type: application/xml
Date: Mon, 12 Nov 2012 23:53:21 GMT
Server: Amazon S3
Content-Length: 231

<Error>
  <Code>InvalidObjectState</Code>
  <Message>The operation is not valid for the object's storage class</Message>
  <RequestId>9FEFFF118E15B86F</RequestId>
  <HostId>WVQ5zkhiT+oiUFDCOiOYv8W4Tk9eNcxWi/MK+hTS/av34XY4rBU3savf0aaaaa</HostId>
</Error>

Sample Response if the Latest Object Is a Delete Marker

Notice that the delete marker returns a 404 Not Found error.

HTTP/1.1 404 Not Found
x-amz-request-id: 318BC8BC148832E5
x-amz-id-2: eftixk72AD6pSkTngzj7UDNNEHGtran
x-amz-version-id: 3GL4kqtJlcpXroDTm3vJBH40Nfr8X8g
x-amz-delete-marker: true
Date: Wed, 28 Oct 2009 22:32:00 GMT
Content-Type: text/plain
Connection: close
Server: AmazonS3

Sample Request: Getting a specified version of an object

The following request returns the specified version of an object.

GET /myObject?versionId=3/L4kqtJlcpXroDTm3vJBH40Nfr8X8g HTTP/1.1
Host: bucket.s3.<Region>.amazonaws.com
Date: Wed, 28 Oct 2009 22:32:00 GMT
Authorization: authorization string
Sample Response: To a versioned object GET request

This example illustrates one usage of GetObject.

HTTP/1.1 200 OK
x-amz-id-2: eftixk72aD6Ap540pIszj7UDNEHGran
x-amz-request-id: 318BC8BC148832E5
Date: Wed, 28 Oct 2009 22:32:00 GMT
Last-Modified: Sun, 1 Jan 2006 12:00:00 GMT
x-amz-version-id: 3/L4kqtJlcpXoDDTmJ+rmSpXd3QBPULUo
ETag: "fba9dedef5f27731c9771645a39863328"
Content-Length: 434234
Content-Type: text/plain
Connection: close
Server: AmazonS3

[434234 bytes of object data]

Sample Request: Parameters altering response header values

The following request specifies all the query string parameters in a GET request overriding the response header values.

GET /Junk3.txt?response-cache-control=No-cache&response-content-disposition=attachment%3B%20filename%3Dtesting.txt&response-content-encoding=x-gzip&response-content-language=mi%2C%20en&response-expires=Thu%2C%2001%20Dec%201994%2016:00:00%20GMT HTTP/1.1
x-amz-date: Sun, 19 Dec 2010 01:53:44 GMT
Accept: */*
Authorization: AWS AKIAIOSFODNN7EXAMPLE:aaStE6nKnw8ihhiIdReoXYl1MamW=

Sample Response: With overridden response header values

The following request specifies all the query string parameters in a GET request overriding the response header values.

HTTP/1.1 200 OK
x-amz-id-2: S1iGWA3hK+Il3/0giu1ZEuegzLAAspsgwawygb9GgFseeFHL5CI8NXSrFwW2
x-amz-request-id: 881B1CBD9DFI7WA1
Date: Sun, 19 Dec 2010 01:54:01 GMT
x-amz-meta-param1: value 1
x-amz-meta-param2: value 2
Cache-Control: No-cache
Content-Language: mi, en
Expires: Thu, 01 Dec 1994 16:00:00 GMT
Content-Disposition: attachment; filename=testing.txt
Content-Encoding: x-gzip
Last-Modified: Fri, 17 Dec 2010 18:10:41 GMT
ETag: "0332beea17bf845f176c5c0d1ae7cf07"
Accept-Ranges: bytes
Content-Type: text/plain
Content-Length: 22
Server: AmazonS3

[object data not shown]
Sample Request: Range header

The following request specifies the HTTP Range header to retrieve the first 10 bytes of an object. For more information about the HTTP Range header, see https://www.w3.org/Protocols/rfc2616/rfc2616-
sec14.html#sec14.35.

Note
Amazon S3 doesn't support retrieving multiple ranges of data per GET request.

GET /example-object HTTP/1.1
Host: example-bucket.s3.<Region>.amazonaws.com
x-amz-date: Fri, 28 Jan 2011 21:32:02 GMT
Range: bytes=0-9
Authorization: AWS AKIAIOSFODNN7EXAMPLE:Yxg83MZaEgh3OZ3l0rLo5RTX11o=
Sample Response with Specified Range of the Object Bytes

Sample Response

In the following sample response, note that the header values are set to the values specified in the true request.

HTTP/1.1 206 Partial Content
x-amz-id-2: MzRISOwyjmnupCzj1WC06l5TTAzm7/JypPGLh0OVFGcJaa03KW/hRaqRoPeEEp
x-amz-request-id: 47622117804B3E11
Date: Fri, 28 Jan 2011 21:32:09 GMT
x-amz-meta-title: the title
Last-Modified: Fri, 28 Jan 2011 20:10:32 GMT
ETag: "b2419b1ef3fd45d596ee22bdf62aaaa2f"
Accept-Ranges: bytes
Content-Range: bytes 0-9/443
Content-Type: text/plain
Content-Length: 10
Server: AmazonS3

[10 bytes of object data]

Sample: Get an object stored using server-side encryption with customer-provided encryption keys

If an object is stored in Amazon S3 using server-side encryption with customer-provided encryption keys, Amazon S3 needs encryption information so that it can decrypt the object before sending it to you in response to a GET request. You provide the encryption information in your GET request using the relevant headers, as shown in the following example request.

GET /example-object HTTP/1.1
Host: example-bucket.s3.<Region>.amazonaws.com
Accept: */*
Authorization: authorization string
Date: Wed, 28 May 2014 19:24:44 +0000
x-amz-server-side-encryption-customer-key:g01CfA30V40jZzSQQJz2ukLRFqtISWorC/8SEKEAMPLE
x-amz-server-side-encryption-customer-key-MD5:ZjQrne1X/iTcskbY2m3example
Sample Response

The following sample response shows some of the response headers Amazon S3 returns. Note that it includes the encryption information in the response.

```
HTTP/1.1 200 OK
x-amz-id-2: ka5jrm8X3N122IY29E989z2tNSJPMcK+to7jMjxImXBbyChqc6tLAy+sau7Vjzh
x-amz-request-id: 195157E3E073D3F9
Date: Wed, 28 May 2014 19:24:45 GMT
Last-Modified: Wed, 28 May 2014 19:21:01 GMT
ETag: "c12022c9a3d63a8d29d90933a2b096"
x-amz-server-side-encryption-customer-algorithm: AES256
x-amz-server-side-encryption-customer-key-MD5: ZjQrne1X/iTcskbY2m3example
```

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3
GetObjectAcl

Service: Amazon Simple Storage Service

Returns the access control list (ACL) of an object. To use this operation, you must have READ_ACP access to the object.

This action is not supported by Amazon S3 on Outposts.

Versioning

By default, GET returns ACL information about the current version of an object. To return ACL information about a different version, use the versionId subresource.

The following operations are related to GetObjectAcl:

- GetObject
- DeleteObject
- PutObject

Request Syntax

```
GET /{Key+}?acl&versionId=VersionId HTTP/1.1
Host: Bucket.s3.amazonaws.com
x-amz-request-payer: RequestPayer
x-amz-expected-bucket-owner: ExpectedBucketOwner
```

URI Request Parameters

The request uses the following URI parameters.

**Bucket (p. 166)**

The bucket name that contains the object for which to get the ACL information.

When using this API with an access point, you must direct requests to the access point hostname. The access point hostname takes the form `AccessPointName-AccountId.s3-accesspoint.Region.amazonaws.com`. When using this operation with an access point through the AWS SDKs, you provide the access point ARN in place of the bucket name. For more information about access point ARNs, see Using Access Points in the Amazon Simple Storage Service Developer Guide.

Required: Yes

**Key (p. 166)**

The key of the object for which to get the ACL information.

Length Constraints: Minimum length of 1.

Required: Yes

**versionId (p. 166)**

VersionId used to reference a specific version of the object.

**x-amz-expected-bucket-owner (p. 166)**

The account id of the expected bucket owner. If the bucket is owned by a different account, the request will fail with an HTTP 403 (Access Denied) error.
x-amz-request-payer (p. 166)

Confirms that the requester knows that they will be charged for the request. Bucket owners need not specify this parameter in their requests. For information about downloading objects from requester pays buckets, see Downloading Objects in Requestor Pays Buckets in the Amazon S3 Developer Guide.

Valid Values: requester

Request Body

The request does not have a request body.

Response Syntax

```xml
HTTP/1.1 200
x-amz-request-charged: RequestCharged
<?xml version="1.0" encoding="UTF-8"?>
<AccessControlPolicy>
    <Owner>
        <DisplayName>string</DisplayName>
        <ID>string</ID>
    </Owner>
    <AccessControlList>
        <Grant>
            <Grantee>
                <DisplayName>string</DisplayName>
                <EmailAddress>string</EmailAddress>
                <ID>string</ID>
                <xsi:type>string</xsi:type>
            </Grantee>
            <Permission>string</Permission>
        </Grant>
    </AccessControlList>
</AccessControlPolicy>
```

Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The response returns the following HTTP headers.

x-amz-request-charged (p. 167)

If present, indicates that the requester was successfully charged for the request.

Valid Values: requester

The following data is returned in XML format by the service.

AccessControlPolicy (p. 167)

Root level tag for the AccessControlPolicy parameters.

Required: Yes

Grants (p. 167)

A list of grants.
Type: Array of Grant (p. 566) data types

Owner (p. 167)

Container for the bucket owner's display name and ID.

Type: Owner (p. 617) data type

The specified key does not exist.

Examples

Sample Request

The following request returns information, including the ACL, of the object my-image.jpg.

```
GET /my-image.jpg?acl HTTP/1.1
Host: bucket.s3.<Region>.amazonaws.com
Date: Wed, 28 Oct 2009 22:32:00 GMT
Authorization: authorization string
```

Sample Response

This example illustrates one usage of GetObjectAcl.

```
HTTP/1.1 200 OK
x-amz-id-2: eftixk72aD6Ap51TnqcoF8eFidJG9Z/2mkiDFu8yU9AS1ed4OpIszj7UDNEHGran
x-amz-request-id: 318BC8BC148832E5
x-amz-version-id: 4HL4kqtJlcpxD5oDTmJz7rM5SpXz3dIbrHY+MTRCxf3vjVfH40Nrfjkd
Date: Wed, 28 Oct 2009 22:32:00 GMT
Last-Modified: Sun, 1 Jan 2006 12:00:00 GMT
Content-Length: 124
Content-Type: text/plain
Connection: close
Server: AmazonS3

<AccessControlPolicy>
   <Owner>
      <ID>75aa57f09aa0c8caea54ba06a</ID>
      <DisplayName>mtd@amazon.com</DisplayName>
   </Owner>
   <AccessControlList>
      <Grant>
         <Grantee xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:type="CanonicalUser">
            <ID>75aa57f09aa0c8caea54ba06a</ID>
            <DisplayName>mtd@amazon.com</DisplayName>
         </Grantee>
         <Permission>FULL_CONTROL</Permission>
      </Grant>
   </AccessControlList>
</AccessControlPolicy>
```

Sample Request: Getting the ACL of the specific version of an object

The following request returns information, including the ACL, of the specified version of the object, my-image.jpg.

```
GET /my-image.jpg?acl HTTP/1.1
Host: bucket.s3.<Region>.amazonaws.com
Date: Wed, 28 Oct 2009 22:32:00 GMT
Authorization: authorization string
```

Sample Response

This example illustrates one usage of GetObjectAcl.
Sample Response: Showing the ACL of the specific version

This example illustrates one usage of GetObjectAcl.

```
<AccessControlPolicy>
  <Owner>
    <ID>75aa57f09aa0c8caeab4f8c24e99d10f8e7faebbf76c078efc7c6cae54ba06a</ID>
    <DisplayName>mdtd@amazon.com</DisplayName>
  </Owner>
  <AccessControlList>
    <Grant>
      <Grantee xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:type="CanonicalUser">
        <ID>75aa57f09aa0c8caeab4f8c24e99d10f8e7faebbf76c078efc7c6cae54ba06a</ID>
      </Grantee>
      <Permission>FULL_CONTROL</Permission>
    </Grant>
  </AccessControlList>
</AccessControlPolicy>
```

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3
GetObjectLegalHold
Service: Amazon Simple Storage Service

Gets an object's current Legal Hold status. For more information, see Locking Objects.

This action is not supported by Amazon S3 on Outposts.

Request Syntax

```
GET /{Key+}?legal-hold&versionId={VersionId} HTTP/1.1
Host: Bucket.s3.amazonaws.com
x-amz-request-payer: Requester
x-amz-expected-bucket-owner: ExpectedBucketOwner
```

URI Request Parameters

The request uses the following URI parameters.

**Bucket (p. 170)**

The bucket name containing the object whose Legal Hold status you want to retrieve.

When using this API with an access point, you must direct requests to the access point hostname. The access point hostname takes the form AccessPointName-AccountId.s3-accesspoint.Region.amazonaws.com. When using this operation with an access point through the AWS SDKs, you provide the access point ARN in place of the bucket name. For more information about access point ARNs, see Using Access Points in the Amazon Simple Storage Service Developer Guide.

Required: Yes

**Key (p. 170)**

The key name for the object whose Legal Hold status you want to retrieve.

Length Constraints: Minimum length of 1.

Required: Yes

**versionId (p. 170)**

The version ID of the object whose Legal Hold status you want to retrieve.

**x-amz-expected-bucket-owner (p. 170)**

The account id of the expected bucket owner. If the bucket is owned by a different account, the request will fail with an HTTP 403 (Access Denied) error.

**x-amz-request-payer (p. 170)**

Confirms that the requester knows that they will be charged for the request. Bucket owners need not specify this parameter in their requests. For information about downloading objects from requester pays buckets, see Downloading Objects in Requestor Pays Buckets in the Amazon S3 Developer Guide.

Valid Values: requester

Request Body

The request does not have a request body.
Response Syntax

HTTP/1.1 200
<?xml version="1.0" encoding="UTF-8"?>
<LegalHold>
  <Status>string</Status>
</LegalHold>

Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in XML format by the service.

LegalHold (p. 171)

Root level tag for the LegalHold parameters.

  Required: Yes

Status (p. 171)

Indicates whether the specified object has a Legal Hold in place.

  Type: String
  Valid Values: ON | OFF

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3
GetObjectLockConfiguration
Service: Amazon Simple Storage Service

Gets the Object Lock configuration for a bucket. The rule specified in the Object Lock configuration will be applied by default to every new object placed in the specified bucket. For more information, see Locking Objects.

Request Syntax

GET /?object-lock HTTP/1.1
Host: Bucket.s3.amazonaws.com
x-amz-expected-bucket-owner: ExpectedBucketOwner

URI Request Parameters

The request uses the following URI parameters.

Bucket (p. 172)

The bucket whose Object Lock configuration you want to retrieve.

When using this API with an access point, you must direct requests to the access point hostname. The access point hostname takes the form AccessPointName-AccountId.s3-accesspoint.Region.amazonaws.com. When using this operation with an access point through the AWS SDKs, you provide the access point ARN in place of the bucket name. For more information about access point ARNs, see Using Access Points in the Amazon Simple Storage Service Developer Guide.

Required: Yes

x-amz-expected-bucket-owner (p. 172)

The account id of the expected bucket owner. If the bucket is owned by a different account, the request will fail with an HTTP 403 (Access Denied) error.

Request Body

The request does not have a request body.

Response Syntax

HTTP/1.1 200
<?xml version="1.0" encoding="UTF-8"?>
<ObjectLockConfiguration>
  <ObjectLockEnabled>string</ObjectLockEnabled>
  <Rule>
    <DefaultRetention>
      <Days>integer</Days>
      <Mode>string</Mode>
      <Years>integer</Years>
    </DefaultRetention>
  </Rule>
</ObjectLockConfiguration>

Response Elements

If the action is successful, the service sends back an HTTP 200 response.

API Version 2006-03-01

172
The following data is returned in XML format by the service.

ObjectLockConfiguration (p. 172)

Root level tag for the ObjectLockConfiguration parameters.

Required: Yes

ObjectLockEnabled (p. 172)

Indicates whether this bucket has an Object Lock configuration enabled.

Type: String

Valid Values: Enabled

Rule (p. 172)

The Object Lock rule in place for the specified object.

Type: ObjectLockRule (p. 612) data type

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3
GetObjectRetention
Service: Amazon Simple Storage Service

Retrieves an object's retention settings. For more information, see Locking Objects.

This action is not supported by Amazon S3 on Outposts.

Request Syntax

```
GET /{Key+}?retention&versionId={VersionId} HTTP/1.1
Host: Bucket.s3.amazonaws.com
x-amz-request-payer: RequestPayer
x-amz-expected-bucket-owner: ExpectedBucketOwner
```

URI Request Parameters

The request uses the following URI parameters.

**Bucket (p. 174)**

The bucket name containing the object whose retention settings you want to retrieve.

When using this API with an access point, you must direct requests to the access point hostname. The access point hostname takes the form `AccessPointName-AccountId.s3-accesspoint.Region.amazonaws.com`. When using this operation with an access point through the AWS SDKs, you provide the access point ARN in place of the bucket name. For more information about access point ARNs, see Using Access Points in the Amazon Simple Storage Service Developer Guide.

Required: Yes

**Key (p. 174)**

The key name for the object whose retention settings you want to retrieve.

Length Constraints: Minimum length of 1.

Required: Yes

**versionId (p. 174)**

The version ID for the object whose retention settings you want to retrieve.

**x-amz-expected-bucket-owner (p. 174)**

The account id of the expected bucket owner. If the bucket is owned by a different account, the request will fail with an HTTP 403 (Access Denied) error.

**x-amz-request-payer (p. 174)**

Confirms that the requester knows that they will be charged for the request. Bucket owners need not specify this parameter in their requests. For information about downloading objects from requester pays buckets, see Downloading Objects in Requestor Pays Buckets in the Amazon S3 Developer Guide.

Valid Values: requester

**Request Body**

The request does not have a request body.
Response Syntax

```xml
HTTP/1.1 200
<?xml version="1.0" encoding="UTF-8"?>
<Retention>
  <Mode>string</Mode>
  <RetainUntilDate>timestamp</RetainUntilDate>
</Retention>
```

Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in XML format by the service.

**Retention (p. 175)**

Root level tag for the Retention parameters.

Required: Yes

**Mode (p. 175)**

Indicates the Retention mode for the specified object.

Type: String

Valid Values: GOVERNANCE | COMPLIANCE

**RetainUntilDate (p. 175)**

The date on which this Object Lock Retention will expire.

Type: Timestamp

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3
GetObjectTagging
Service: Amazon Simple Storage Service

Returns the tag-set of an object. You send the GET request against the tagging subresource associated with the object.

To use this operation, you must have permission to perform the `s3:GetObjectTagging` action. By default, the GET operation returns information about current version of an object. For a versioned bucket, you can have multiple versions of an object in your bucket. To retrieve tags of any other version, use the `versionId` query parameter. You also need permission for the `s3:GetObjectVersionTagging` action.

By default, the bucket owner has this permission and can grant this permission to others.

For information about the Amazon S3 object tagging feature, see Object Tagging.

The following operation is related to GetObjectTagging:

- PutObjectTagging

Request Syntax

```
GET /{Key+}?tagging&versionId=VersionId HTTP/1.1
Host: Bucket.s3.amazonaws.com
x-amz-expected-bucket-owner: ExpectedBucketOwner
```

URI Request Parameters

The request uses the following URI parameters.

**Bucket (p. 176)**

The bucket name containing the object for which to get the tagging information.

When using this API with an access point, you must direct requests to the access point hostname. The access point hostname takes the form `AccessPointName-AccountId.s3-accesspoint.Region.amazonaws.com`. When using this operation with an access point through the AWS SDKs, you provide the access point ARN in place of the bucket name. For more information about access point ARNs, see Using Access Points in the Amazon Simple Storage Service Developer Guide.

When using this API with Amazon S3 on Outposts, you must direct requests to the S3 on Outposts hostname. The S3 on Outposts hostname takes the form `AccessPointName-AccountId.outpostID.s3-outposts.Region.amazonaws.com`. When using this operation using S3 on Outposts through the AWS SDKs, you provide the Outposts bucket ARN in place of the bucket name. For more information about S3 on Outposts ARNs, see Using S3 on Outposts in the Amazon Simple Storage Service Developer Guide.

Required: Yes

**Key (p. 176)**

Object key for which to get the tagging information.

Length Constraints: Minimum length of 1.

Required: Yes
versionId (p. 176)

The versionId of the object for which to get the tagging information.

x-amz-expected-bucket-owner (p. 176)

The account id of the expected bucket owner. If the bucket is owned by a different account, the request will fail with an HTTP 403 (Access Denied) error.

Request Body

The request does not have a request body.

Response Syntax

```
HTTP/1.1 200
x-amz-version-id: VersionId
<?xml version="1.0" encoding="UTF-8"?>
<Tagging>
  <TagSet>
    <Tag>
      <Key>string</Key>
      <Value>string</Value>
    </Tag>
  </TagSet>
</Tagging>
```

Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The response returns the following HTTP headers.

x-amz-version-id (p. 177)

The versionId of the object for which you got the tagging information.

The following data is returned in XML format by the service.

Tagging (p. 177)

Root level tag for the Tagging parameters.

Required: Yes

TagSet (p. 177)

Contains the tag set.

Type: Array of Tag (p. 666) data types

Examples

Sample Request

The following request returns the tag set of the specified object.

```
GET /example-object?tagging HTTP/1.1
```

API Version 2006-03-01

177
Sample Response

This example illustrates one usage of GetObjectTagging.

```
HTTP/1.1 200 OK
Date: Thu, 22 Sep 2016 21:33:08 GMT
Server: AmazonS3
<?xml version="1.0" encoding="UTF-8"?>
<Tagging xmlns="http://s3.amazonaws.com/doc/2006-03-01/">
  <TagSet>
    <Tag>
      <Key>tag1</Key>
      <Value>val1</Value>
    </Tag>
    <Tag>
      <Key>tag2</Key>
      <Value>val2</Value>
    </Tag>
  </TagSet>
</Tagging>
```

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3
GetObjectTorrent
Service: Amazon Simple Storage Service

Returns torrent files from a bucket. BitTorrent can save you bandwidth when you're distributing large files. For more information about BitTorrent, see Using BitTorrent with Amazon S3.

Note
You can get torrent only for objects that are less than 5 GB in size, and that are not encrypted using server-side encryption with a customer-provided encryption key.

To use GET, you must have READ access to the object.

This action is not supported by Amazon S3 on Outposts.

The following operation is related to GetObjectTorrent:

- GetObject

Request Syntax

```
GET /{Key+}?torrent HTTP/1.1
Host: Bucket.s3.amazonaws.com
x-amz-request-payer: RequestPayer
x-amz-expected-bucket-owner: ExpectedBucketOwner
```

URI Request Parameters

The request uses the following URI parameters.

Bucket (p. 179)

The name of the bucket containing the object for which to get the torrent files.

Required: Yes

Key (p. 179)

The object key for which to get the information.

Length Constraints: Minimum length of 1.

Required: Yes

x-amz-expected-bucket-owner (p. 179)

The account id of the expected bucket owner. If the bucket is owned by a different account, the request will fail with an HTTP 403 (Access Denied) error.

x-amz-request-payer (p. 179)

Confirms that the requester knows that they will be charged for the request. Bucket owners need not specify this parameter in their requests. For information about downloading objects from requester pays buckets, see Downloading Objects in Requestor Pays Buckets in the Amazon S3 Developer Guide.

Valid Values: requester

Request Body

The request does not have a request body.
Response Syntax

HTTP/1.1 200
x-amz-request-charged: RequestCharged

Body

Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The response returns the following HTTP headers.

x-amz-request-charged (p. 180)

If present, indicates that the requester was successfully charged for the request.

Valid Values: requester

The following data is returned in binary format by the service.

Examples

Getting torrent files in a bucket

This example retrieves the Torrent file for the Nelson object in the quotes bucket.

```
GET /quotes/Nelson?torrent HTTP/1.0
Host: bucket.s3.<Region>.amazonaws.com
Date: Wed, 28 Oct 2009 22:32:00 GMT
Authorization: authorization string
```

Sample Response

This example illustrates one usage of GetObjectTorrent.

```
HTTP/1.1 200 OK
x-amz-request-id: 7CD745EBB7AB5ED9
Date: Wed, 25 Nov 2009 12:00:00 GMT
Content-Disposition: attachment; filename=Nelson.torrent;
Content-Type: application/x-bittorrent
Content-Length: 537
Server: AmazonS3

<body: a Bencoded dictionary as defined by the BitTorrent specification>
```

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
• AWS SDK for C++
• AWS SDK for Go
• AWS SDK for Java
• AWS SDK for JavaScript
• AWS SDK for PHP V3
• AWS SDK for Python
• AWS SDK for Ruby V3
GetPublicAccessBlock
Service: Amazon Simple Storage Service

Retrieves the PublicAccessBlock configuration for an Amazon S3 bucket. To use this operation, you must have the s3:GetBucketPublicAccessBlock permission. For more information about Amazon S3 permissions, see Specifying Permissions in a Policy.

**Important**
When Amazon S3 evaluates the PublicAccessBlock configuration for a bucket or an object, it checks the PublicAccessBlock configuration for both the bucket (or the bucket that contains the object) and the bucket owner's account. If the PublicAccessBlock settings are different between the bucket and the account, Amazon S3 uses the most restrictive combination of the bucket-level and account-level settings.

For more information about when Amazon S3 considers a bucket or an object public, see The Meaning of "Public".

The following operations are related to GetPublicAccessBlock:
- Using Amazon S3 Block Public Access
- PutPublicAccessBlock
- GetPublicAccessBlock
- DeletePublicAccessBlock

**Request Syntax**

```
GET /?publicAccessBlock HTTP/1.1
Host: Bucket.s3.amazonaws.com
x-amz-expected-bucket-owner: ExpectedBucketOwner
```

**URI Request Parameters**

The request uses the following URI parameters.

**Bucket (p. 182)**

The name of the Amazon S3 bucket whose PublicAccessBlock configuration you want to retrieve.

Required: Yes

**x-amz-expected-bucket-owner (p. 182)**

The account id of the expected bucket owner. If the bucket is owned by a different account, the request will fail with an HTTP 403 (Access Denied) error.

**Request Body**

The request does not have a request body.

**Response Syntax**

```
HTTP/1.1 200
<?xml version="1.0" encoding="UTF-8"?>
<PublicAccessBlockConfiguration>
  <BlockPublicAcls>boolean</BlockPublicAcls>
</PublicAccessBlockConfiguration>
```
Amazon Simple Storage Service API Reference
Amazon Simple Storage Service

```xml
<IgnorePublicAcls>boolean</IgnorePublicAcls>
<BlockPublicPolicy>boolean</BlockPublicPolicy>
<RestrictPublicBuckets>boolean</RestrictPublicBuckets>
</PublicAccessBlockConfiguration>
```

### Response Elements

If the action is successful, the service sends back an HTTP 200 response. The following data is returned in XML format by the service.

**PublicAccessBlockConfiguration (p. 182)**

Root level tag for the PublicAccessBlockConfiguration parameters.

- Required: Yes

**BlockPublicAcls (p. 182)**

Specifies whether Amazon S3 should block public access control lists (ACLs) for this bucket and objects in this bucket. Setting this element to `true` causes the following behavior:
- PUT Bucket acl and PUT Object acl calls fail if the specified ACL is public.
- PUT Object calls fail if the request includes a public ACL.
- PUT Bucket calls fail if the request includes a public ACL.

Enabling this setting doesn't affect existing policies or ACLs.

- Type: Boolean

**BlockPublicPolicy (p. 182)**

Specifies whether Amazon S3 should block public bucket policies for this bucket. Setting this element to `true` causes Amazon S3 to reject calls to PUT Bucket policy if the specified bucket policy allows public access.

Enabling this setting doesn't affect existing bucket policies.

- Type: Boolean

**IgnorePublicAcls (p. 182)**

Specifies whether Amazon S3 should ignore public ACLs for this bucket and objects in this bucket. Setting this element to `true` causes Amazon S3 to ignore all public ACLs on this bucket and objects in this bucket.

Enabling this setting doesn't affect the persistence of any existing ACLs and doesn't prevent new public ACLs from being set.

- Type: Boolean

**RestrictPublicBuckets (p. 182)**

Specifies whether Amazon S3 should restrict public bucket policies for this bucket. Setting this element to `true` restricts access to this bucket to only AWS service principals and authorized users within this account if the bucket has a public policy.

Enabling this setting doesn't affect previously stored bucket policies, except that public and cross-account access within any public bucket policy, including non-public delegation to specific accounts, is blocked.

- Type: Boolean
Examples

Sample Request

The following request gets a bucket PublicAccessBlock configuration.

```
GET /<bucket-name>?publicAccessBlock HTTP/1.1
Host: <bucket-name>.s3.<Region>.amazonaws.com
x-amz-date: <Thu, 15 Nov 2016 00:17:21 GMT>
Authorization: <signatureValue>
```

Sample Response

This example illustrates one usage of GetPublicAccessBlock.

```
HTTP/1.1 200 OK
x-amz-id-2: ITnGT1y4REXAMPLEPi4hklTXouTf0hccUjo0iCPEXAMPLEutBj3M7fPGlW02SEWp
x-amz-request-id: 51991EXAMPLE5321
Date: Thu, 15 Nov 2016 00:17:22 GMT
Server: AmazonS3
Content-Length: 0

<PublicAccessBlockConfiguration>
  <BlockPublicAcls>TRUE</BlockPublicAcls>
  <IgnorePublicAcls>FALSE</IgnorePublicAcls>
  <BlockPublicPolicy>FALSE</BlockPublicPolicy>
  <RestrictPublicBuckets>FALSE</RestrictPublicBuckets>
</PublicAccessBlockConfiguration>
```

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3
**HeadBucket**  
**Service: Amazon Simple Storage Service**

This operation is useful to determine if a bucket exists and you have permission to access it. The operation returns a **200 OK** if the bucket exists and you have permission to access it. Otherwise, the operation might return responses such as **404 Not Found** and **403 Forbidden**.

To use this operation, you must have permissions to perform the **s3:ListBucket** action. The bucket owner has this permission by default and can grant this permission to others. For more information about permissions, see Permissions Related to Bucket Subresource Operations and Managing Access Permissions to Your Amazon S3 Resources.

**Request Syntax**

```
HEAD / HTTP/1.1
Host: Bucket.s3.amazonaws.com
x-amz-expected-bucket-owner: ExpectedBucketOwner
```

**URI Request Parameters**

The request uses the following URI parameters.

**Bucket (p. 185)**

The bucket name.

When using this API with an access point, you must direct requests to the access point hostname. The access point hostname takes the form `AccessPointName-AccountId.s3-accesspoint.Region.amazonaws.com`. When using this operation with an access point through the AWS SDKs, you provide the access point ARN in place of the bucket name. For more information about access point ARNs, see Using Access Points in the Amazon Simple Storage Service Developer Guide.

When using this API with Amazon S3 on Outposts, you must direct requests to the S3 on Outposts hostname. The S3 on Outposts hostname takes the form `AccessPointName-AccountId.outpostId.s3-outposts.Region.amazonaws.com`. When using this operation using S3 on Outposts through the AWS SDKs, you provide the Outposts bucket ARN in place of the bucket name. For more information about S3 on Outposts ARNs, see Using S3 on Outposts in the Amazon Simple Storage Service Developer Guide.

Required: Yes

**x-amz-expected-bucket-owner (p. 185)**

The account id of the expected bucket owner. If the bucket is owned by a different account, the request will fail with an HTTP **403 (Access Denied)** error.

**Request Body**

The request does not have a request body.

**Response Syntax**

```
HTTP/1.1 200
```

**Response Elements**

If the action is successful, the service sends back an HTTP 200 response with an empty HTTP body.
The specified bucket does not exist.

Examples

Sample Request

This example illustrates one usage of HeadBucket.

```
HEAD / HTTP/1.1
Date: Fri, 10 Feb 2012 21:34:55 GMT
Authorization: authorization string
Host: myawsbucket.s3.amazonaws.com
Connection: Keep-Alive
```

Sample Response

This example illustrates one usage of HeadBucket.

```
HTTP/1.1 200 OK
x-amz-id-2: JuKZqmXuiwFeDQxhD7MSKtsKobSzWA1QEjLbTMTagkRdBX2z71L/jGhDeJ3j6s80
x-amz-request-id: 32FE2CEB32F5EE25
x-amz-bucket-region: us-west-2
Date: Fri, 10 2012 21:34:56 GMT
Server: AmazonS3
```

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3
HeadObject
Service: Amazon Simple Storage Service

The HEAD operation retrieves metadata from an object without returning the object itself. This operation is useful if you're only interested in an object's metadata. To use HEAD, you must have READ access to the object.

A HEAD request has the same options as a GET operation on an object. The response is identical to the GET response except that there is no response body.

If you encrypt an object by using server-side encryption with customer-provided encryption keys (SSE-C) when you store the object in Amazon S3, then when you retrieve the metadata from the object, you must use the following headers:

- x-amz-server-side-encryption-customer-algorithm
- x-amz-server-side-encryption-customer-key
- x-amz-server-side-encryption-customer-key-MD5

For more information about SSE-C, see Server-Side Encryption (Using Customer-Provided Encryption Keys).

Note
Encryption request headers, like x-amz-server-side-encryption, should not be sent for GET requests if your object uses server-side encryption with CMKs stored in AWS KMS (SSE-KMS) or server-side encryption with Amazon S3–managed encryption keys (SSE-S3). If your object does use these types of keys, you'll get an HTTP 400 BadRequest error.

Request headers are limited to 8 KB in size. For more information, see Common Request Headers.

Consider the following when using request headers:

- Consideration 1 – If both of the If-Match and If-Unmodified-Since headers are present in the request as follows:
  - If-Match condition evaluates to true, and;
  - If-Unmodified-Since condition evaluates to false;
  Then Amazon S3 returns 200 OK and the data requested.
- Consideration 2 – If both of the If-None-Match and If-Modified-Since headers are present in the request as follows:
  - If-None-Match condition evaluates to false, and;
  - If-Modified-Since condition evaluates to true;
  Then Amazon S3 returns the 304 Not Modified response code.

For more information about conditional requests, see RFC 7232.

Permissions
You need the s3:GetObject permission for this operation. For more information, see Specifying Permissions in a Policy. If the object you request does not exist, the error Amazon S3 returns depends on whether you also have the s3:ListBucket permission.

- If you have the s3:ListBucket permission on the bucket, Amazon S3 returns an HTTP status code 404 ("no such key") error.
- If you don't have the s3:ListBucket permission, Amazon S3 returns an HTTP status code 403 ("access denied") error.
The following operation is related to HeadObject:

- GetObject

Request Syntax

```
HEAD /Key+?partNumber=PartNumber&versionId=VersionId HTTP/1.1
Host: Bucket.s3.amazonaws.com
If-Match: IfMatch
If-Modified-Since: IfModifiedSince
If-None-Match: IfNoneMatch
If-Unmodified-Since: IfUnmodifiedSince
Range: Range
x-amz-server-side-encryption-customer-algorithm: SSECustomerAlgorithm
x-amz-server-side-encryption-customer-key: SSECustomerKey
x-amz-server-side-encryption-customer-key-MD5: SSECustomerKeyMD5
x-amz-request-payer: RequestPayer
x-amz-expected-bucket-owner: ExpectedBucketOwner
```

URI Request Parameters

The request uses the following URI parameters.

**Bucket (p. 188)**

The name of the bucket containing the object.

When using this API with an access point, you must direct requests to the access point hostname. The access point hostname takes the form `AccessPointName-AccountId.s3-accesspoint.Region.amazonaws.com`. When using this operation with an access point through the AWS SDKs, you provide the access point ARN in place of the bucket name. For more information about access point ARNs, see Using Access Points in the Amazon Simple Storage Service Developer Guide.

When using this API with Amazon S3 on Outposts, you must direct requests to the S3 on Outposts hostname. The S3 on Outposts hostname takes the form `AccessPointName-AccountId.outpostId.s3-outposts.Region.amazonaws.com`. When using this operation using S3 on Outposts through the AWS SDKs, you provide the Outposts bucket ARN in place of the bucket name. For more information about S3 on Outposts ARNs, see Using S3 on Outposts in the Amazon Simple Storage Service Developer Guide.

Required: Yes

**If-Match (p. 188)**

Return the object only if its entity tag (ETag) is the same as the one specified, otherwise return a 412 (precondition failed).

**If-Modified-Since (p. 188)**

Return the object only if it has been modified since the specified time, otherwise return a 304 (not modified).

**If-None-Match (p. 188)**

Return the object only if its entity tag (ETag) is different from the one specified, otherwise return a 304 (not modified).

**If-Unmodified-Since (p. 188)**

Return the object only if it has not been modified since the specified time, otherwise return a 412 (precondition failed).
Key (p. 188)

The object key.

Length Constraints: Minimum length of 1.

Required: Yes

partNumber (p. 188)

Part number of the object being read. This is a positive integer between 1 and 10,000. Effectively performs a 'ranged' HEAD request for the part specified. Useful querying about the size of the part and the number of parts in this object.

Range (p. 188)

Downloads the specified range bytes of an object. For more information about the HTTP Range header, see http://www.w3.org/Protocols/rfc2616/rfc2616-sec14.html#sec14.35.

Note
Amazon S3 doesn't support retrieving multiple ranges of data per GET request.

versionId (p. 188)

VersionId used to reference a specific version of the object.

x-amz-expected-bucket-owner (p. 188)

The account id of the expected bucket owner. If the bucket is owned by a different account, the request will fail with an HTTP 403 (Access Denied) error.

x-amz-request-payer (p. 188)

Confirms that the requester knows that they will be charged for the request. Bucket owners need not specify this parameter in their requests. For information about downloading objects from requester pays buckets, see Downloading Objects in Requestor Pays Buckets in the Amazon S3 Developer Guide.

Valid Values: requester

x-amz-server-side-encryption-customer-algorithm (p. 188)

Specifies the algorithm to use when encrypting the object (for example, AES256).

x-amz-server-side-encryption-customer-key (p. 188)

Specifies the customer-provided encryption key for Amazon S3 to use in encrypting data. This value is used to store the object and then it is discarded; Amazon S3 does not store the encryption key. The key must be appropriate for use with the algorithm specified in the x-amz-server-side-encryption-customer-algorithm header.

x-amz-server-side-encryption-customer-key-MD5 (p. 188)

Specifies the 128-bit MD5 digest of the encryption key according to RFC 1321. Amazon S3 uses this header for a message integrity check to ensure that the encryption key was transmitted without error.

Request Body

The request does not have a request body.

Response Syntax

```
HTTP/1.1 200
x-amz-delete-marker: DeleteMarker
```
<table>
<thead>
<tr>
<th>Response Elements</th>
</tr>
</thead>
<tbody>
<tr>
<td>If the action is successful, the service sends back an HTTP 200 response.</td>
</tr>
</tbody>
</table>

The response returns the following HTTP headers.

<table>
<thead>
<tr>
<th>accept-ranges (p. 189)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Indicates that a range of bytes was specified.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Cache-Control (p. 189)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Specifies caching behavior along the request/reply chain.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Content-Disposition (p. 189)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Specifies presentational information for the object.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Content-Encoding (p. 189)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Specifies what content encodings have been applied to the object and thus what decoding mechanisms must be applied to obtain the media-type referenced by the Content-Type header field.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Content-Language (p. 189)</th>
</tr>
</thead>
<tbody>
<tr>
<td>The language the content is in.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Content-Length (p. 189)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Size of the body in bytes.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Content-Type (p. 189)</th>
</tr>
</thead>
<tbody>
<tr>
<td>A standard MIME type describing the format of the object data.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>ETag (p. 189)</th>
</tr>
</thead>
<tbody>
<tr>
<td>An ETag is an opaque identifier assigned by a web server to a specific version of a resource found at a URL.</td>
</tr>
</tbody>
</table>
Expires (p. 189)

The date and time at which the object is no longer cacheable.

Last-Modified (p. 189)

Last modified date of the object

x-amz-archive-status (p. 189)

The archive state of the head object.

Valid Values: ARCHIVE_ACCESS | DEEP_ARCHIVE_ACCESS

x-amz-delete-marker (p. 189)

Specifies whether the object retrieved was (true) or was not (false) a Delete Marker. If false, this response header does not appear in the response.

x-amz-expiration (p. 189)

If the object expiration is configured (see PUT Bucket lifecycle), the response includes this header. It includes the expiry-date and rule-id key-value pairs providing object expiration information. The value of the rule-id is URL encoded.

x-amz-missing-meta (p. 189)

This is set to the number of metadata entries not returned in x-amz-meta headers. This can happen if you create metadata using an API like SOAP that supports more flexible metadata than the REST API. For example, using SOAP, you can create metadata whose values are not legal HTTP headers.

x-amz-mp-parts-count (p. 189)

The count of parts this object has.

x-amz-object-lock-legal-hold (p. 189)

Specifies whether a legal hold is in effect for this object. This header is only returned if the requester has the s3:GetObjectLegalHold permission. This header is not returned if the specified version of this object has never had a legal hold applied. For more information about S3 Object Lock, see Object Lock.

Valid Values: ON | OFF

x-amz-object-lock-mode (p. 189)

The Object Lock mode, if any, that's in effect for this object. This header is only returned if the requester has the s3:GetObjectRetention permission. For more information about S3 Object Lock, see Object Lock.

Valid Values: GOVERNANCE | COMPLIANCE

x-amz-object-lock-retain-until-date (p. 189)

The date and time when the Object Lock retention period expires. This header is only returned if the requester has the s3:GetObjectRetention permission.

x-amz-replication-status (p. 189)

Amazon S3 can return this header if your request involves a bucket that is either a source or destination in a replication rule.

In replication, you have a source bucket on which you configure replication and destination bucket where Amazon S3 stores object replicas. When you request an object (GetObject) or object metadata (HeadObject) from these buckets, Amazon S3 will return the x-amz-replication-status header in the response as follows:

- If requesting an object from the source bucket — Amazon S3 will return the x-amz-replication-status header if the object in your request is eligible for replication.
For example, suppose that in your replication configuration, you specify object prefix `TaxDocs` requesting Amazon S3 to replicate objects with key prefix `TaxDocs`. Any objects you upload with this key name prefix, for example `TaxDocs/document1.pdf`, are eligible for replication. For any object request with this key name prefix, Amazon S3 will return the `x-amz-replication-status` header with value PENDING, COMPLETED or FAILED indicating object replication status.

- If requesting an object from the destination bucket — Amazon S3 will return the `x-amz-replication-status` header with value REPLICA if the object in your request is a replica that Amazon S3 created.

For more information, see Replication.

Valid Values: COMPLETE | PENDING | FAILED | REPLICA

`x-amz-request-charged (p. 189)`

If present, indicates that the requester was successfully charged for the request.

Valid Values: requester

`x-amz-restore (p. 189)`

If the object is an archived object (an object whose storage class is GLACIER), the response includes this header if either the archive restoration is in progress (see RestoreObject or an archive copy is already restored.

If an archive copy is already restored, the header value indicates when Amazon S3 is scheduled to delete the object copy. For example:

`x-amz-restore: ongoing-request="false", expiry-date="Fri, 23 Dec 2012 00:00:00 GMT"`

If the object restoration is in progress, the header returns the value ongoing-request="true".

For more information about archiving objects, see Transitioning Objects: General Considerations.

`x-amz-server-side-encryption (p. 189)`

If the object is stored using server-side encryption either with an AWS KMS customer master key (CMK) or an Amazon S3-managed encryption key, the response includes this header with the value of the server-side encryption algorithm used when storing this object in Amazon S3 (for example, AES256, aws:kms).

Valid Values: AES256 | aws:kms

`x-amz-server-side-encryption-aws-kms-key-id (p. 189)`

If present, specifies the ID of the AWS Key Management Service (AWS KMS) symmetric customer managed customer master key (CMK) that was used for the object.

`x-amz-server-side-encryption-customer-algorithm (p. 189)`

If server-side encryption with a customer-provided encryption key was requested, the response will include this header confirming the encryption algorithm used.

`x-amz-server-side-encryption-customer-key-MD5 (p. 189)`

If server-side encryption with a customer-provided encryption key was requested, the response will include this header to provide round-trip message integrity verification of the customer-provided encryption key.

`x-amz-storage-class (p. 189)`

Provides storage class information of the object. Amazon S3 returns this header for all objects except for S3 Standard storage class objects.
For more information, see Storage Classes.

Valid Values: STANDARD | REDUCED_REDUNDANCY | STANDARD_IA | ONEZONE_IA | INTELLIGENT_TIERING | GLACIER | DEEP_ARCHIVE | OUTPOSTS

x-amz-version-id (p. 189)

Version of the object.

x-amz-website-redirect-location (p. 189)

If the bucket is configured as a website, redirects requests for this object to another object in the same bucket or to an external URL. Amazon S3 stores the value of this header in the object metadata.

The specified key does not exist.

Examples

Sample Request

The following request returns the metadata of an object.

```
HEAD /my-image.jpg HTTP/1.1
Host: bucket.s3.<Region>.amazonaws.com
Date: Wed, 28 Oct 2009 22:32:00 GMT
Authorization: AWS AKIAIOSFODNN7EXAMPLE:02236Q3V0RonhpaBX5sCYVfibNRuU=
```

Sample Response

This example illustrates one usage of HeadObject.

```
HTTP/1.1 200 OK
x-amz-id-2: ef8yU9AS1ed4Opiszj7UDNEHGran
x-amz-request-id: 318BC8BC143432E5
x-amz-version-id: 3HL4kqtJlcpXroDThmjVBJH4ORjfkd
Date: Wed, 28 Oct 2009 22:32:00 GMT
Last-Modified: Sun, 1 Jan 2006 12:00:00 GMT
ETag: "fba9dede5f277731c9771645a39863328"
Content-Length: 434234
Content-Type: text/plain
Connection: close
Server: AmazonS3
```

Sample Response: With an expiration tag

If the object is scheduled to expire according to a lifecycle configuration set on the bucket, the response returns the x-amz-expiration tag with information about when Amazon S3 will delete the object. For more information, see Transitioning Objects: General Considerations.

```
HTTP/1.1 200 OK
x-amz-id-2: azQRZtQJ2m1P8R+TIsG9h0VuC/DmiSjMjXUMq7snk+LKSJeurtmfzS1Gr46GzSJ
x-amz-request-id: 0EF61CCEF24A26
Date: Mon, 17 Dec 2012 02:26:39 GMT
Last-Modified: Mon, 17 Dec 2012 02:14:10 GMT
```
Sample Request: Getting metadata from a specified version of an object

The following request returns the metadata of the specified version of an object.

```
HEAD /my-image.jpg?versionId=3HL4kqCxf3vjVBH40Nrjfkd HTTP/1.1
Host: bucket.s3.<Region>.amazonaws.com
Date: Wed, 28 Oct 2009 22:32:00 GMT
Authorization: AWS AKIAIOSFODNN7EXAMPLE:02236Q3V0WpaBX5sCYVf1bNRuU=
```

Sample Response: To a versioned HEAD request

This example illustrates one usage of HeadObject.

```
HTTP/1.1 200 OK
x-amz-id-2: eftixk72aD6Ap51TnqcoF8epI.szj7UDNEHGran
x-amz-request-id: 318BC8BC143432E5
x-amz-version-id: 3HL4kqtJlcpXrof3vjVBH40Nrjfkd
Date: Wed, 28 Oct 2009 22:30:00 GMT
Last-Modified: Sun, 1 Jan 2006 12:00:00 GMT
ETag: "fba9dede5f27731c9771645a39863328"
Content-Length: 434234
Content-Type: text/plain
Connection: close
Server: AmazonS3
```

Sample Request: For a S3 Glacier object

For an archived object, the `x-amz-restore` header provides the date when the restored copy expires, as shown in the following response. Even if the object is stored in Glacier, all object metadata is still available.

```
HEAD /my-image.jpg HTTP/1.1
Host: bucket.s3.<Region>.amazonaws.com
Date: 13 Nov 2012 00:28:38 GMT
Authorization: AWS AKIAIOSFODNN7EXAMPLE:02236Q3V0RonhpaBX5sCYVf1bNRuU=
```

Sample Response: S3 Glacier object

If the object is already restored, the `x-amz-restore` header provides the date when the restored copy will expire, as shown in the following response.

```
HTTP/1.1 200 OK
x-amz-id-2: FSVaTmjrmb3l1zslNnWBZeM7M191I8UBxMini0A8AirHANJBo+hEftBu1ESACOMJp
x-amz-request-id: E5CEFBC143EB505A
```
Sample Response: In-progress restoration

If the restoration is in progress, the x-amz-restore header returns a message accordingly.

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3
ListBucketAnalyticsConfigurations
Service: Amazon Simple Storage Service

Lists the analytics configurations for the bucket. You can have up to 1,000 analytics configurations per bucket.

This operation supports list pagination and does not return more than 100 configurations at a time. You should always check the IsTruncated element in the response. If there are no more configurations to list, IsTruncated is set to false. If there are more configurations to list, IsTruncated is set to true, and there will be a value in NextContinuationToken. You use the NextContinuationToken value to continue the pagination of the list by passing the value in continuation-token in the request to GET the next page.

To use this operation, you must have permissions to perform the s3:GetAnalyticsConfiguration action. The bucket owner has this permission by default. The bucket owner can grant this permission to others. For more information about permissions, see Permissions Related to Bucket Subresource Operations and Managing Access Permissions to Your Amazon S3 Resources.

For information about Amazon S3 analytics feature, see Amazon S3 Analytics – Storage Class Analysis.

The following operations are related to ListBucketAnalyticsConfigurations:

- GetBucketAnalyticsConfiguration
- DeleteBucketAnalyticsConfiguration
- PutBucketAnalyticsConfiguration

Request Syntax

```
GET /?analytics&continuation-token=ContinuationToken HTTP/1.1
Host: Bucket.s3.amazonaws.com
x-amz-expected-bucket-owner: ExpectedBucketOwner
```

URI Request Parameters

The request uses the following URI parameters.

Bucket (p. 196)

The name of the bucket from which analytics configurations are retrieved.

Required: Yes

continuation-token (p. 196)

The ContinuationToken that represents a placeholder from where this request should begin.

x-amz-expected-bucket-owner (p. 196)

The account id of the expected bucket owner. If the bucket is owned by a different account, the request will fail with an HTTP 403 (Access Denied) error.

Request Body

The request does not have a request body.

Response Syntax

```
HTTP/1.1 200
```
<ListBucketAnalyticsConfigurationResult>
  <IsTruncated>boolean</IsTruncated>
  <ContinuationToken>string</ContinuationToken>
  <NextContinuationToken>string</NextContinuationToken>
  <AnalyticsConfiguration>
    <Filter>
      <And>
        <Prefix>string</Prefix>
        <Tag>
          <Key>string</Key>
          <Value>string</Value>
        </Tag>
        ...
      </And>
      <Prefix>string</Prefix>
      <Tag>
        <Key>string</Key>
        <Value>string</Value>
      </Tag>
    </Filter>
    <Id>string</Id>
    <StorageClassAnalysis>
      <DataExport>
        <Destination>
          <S3BucketDestination>
            <Bucket>string</Bucket>
            <BucketAccountId>string</BucketAccountId>
            <Format>string</Format>
            <Prefix>string</Prefix>
          </S3BucketDestination>
        </Destination>
      </DataExport>
    </StorageClassAnalysis>
    ...
  </AnalyticsConfiguration>
</ListBucketAnalyticsConfigurationResult>

Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in XML format by the service.

ListBucketAnalyticsConfigurationResult (p. 196)

Root level tag for the ListBucketAnalyticsConfigurationResult parameters.

Required: Yes

AnalyticsConfiguration (p. 196)

The list of analytics configurations for a bucket.

Type: Array of AnalyticsConfiguration (p. 519) data types

ContinuationToken (p. 196)

The marker that is used as a starting point for this analytics configuration list response. This value is present if it was sent in the request.

Type: String
IsTruncated (p. 196)

Indicates whether the returned list of analytics configurations is complete. A value of true indicates that the list is not complete and the NextContinuationToken will be provided for a subsequent request.

Type: Boolean

NextContinuationToken (p. 196)

NextContinuationToken is sent when IsTruncated is true, which indicates that there are more analytics configurations to list. The next request must include this NextContinuationToken. The token is obfuscated and is not a usable value.

Type: String

Examples

Sample Request

Delete the metric configuration with a specified ID, which disables the CloudWatch metrics with the ExampleMetrics value for the FilterId dimension.

```
GET /?analytics HTTP/1.1
Host: example-bucket.s3.<Region>.amazonaws.com
x-amz-date: 20160430T233541Z
Authorization: authorization string
```

Sample Response

This example illustrates one usage of ListBucketAnalyticsConfigurations.

```
HTTP/1.1 200 OK
x-amz-id-2: gyB+3jRPnrkN98ZajxHXr3u7EFM67bNgSAxexeEHndCX/7GRnfTXxReKUQF28IfP
x-amz-request-id: 3B3C7C725673C630
Date: Sat, 30 Apr 2016 23:29:37 GMT
Content-Length: length
Server: AmazonS3

<ListBucketAnalyticsConfigurationResult xmlns="http://s3.amazonaws.com/doc/2006-03-01/">
  <AnalyticsConfiguration>
    <Id>list1</Id>
    <Filter>
      <And>
        <Prefix>images/</Prefix>
        <Tag>
          <Key>dog</Key>
          <Value>corgi</Value>
        </Tag>
      </And>
    </Filter>
    <StorageClassAnalysis>
      <DataExport>
        <OutputSchemaVersion>V_1</OutputSchemaVersion>
        <Destination>
          <S3BucketDestination>
            <Format>CSV</Format>
            <BucketAccountId>123456789012</BucketAccountId>
          </S3BucketDestination>
        </Destination>
      </DataExport>
    </StorageClassAnalysis>
  </AnalyticsConfiguration>
</ListBucketAnalyticsConfigurationResult>
```
See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3
ListBucketIntelligentTieringConfigurations
Service: Amazon Simple Storage Service

Lists the S3 Intelligent-Tiering configuration from the specified bucket.

The S3 Intelligent-Tiering storage class is designed to optimize storage costs by automatically moving
data to the most cost-effective storage access tier, without additional operational overhead. S3
Intelligent-Tiering delivers automatic cost savings by moving data between access tiers, when access
patterns change.

The S3 Intelligent-Tiering storage class is suitable for objects larger than 128 KB that you plan to store
for at least 30 days. If the size of an object is less than 128 KB, it is not eligible for auto-tiering. Smaller
objects can be stored, but they are always charged at the frequent access tier rates in the S3 Intelligent-
Tiering storage class.

If you delete an object before the end of the 30-day minimum storage duration period, you are charged
for 30 days. For more information, see Storage class for automatically optimizing frequently and
infrequently accessed objects.

Operations related to ListBucketIntelligentTieringConfigurations include:

- DeleteBucketIntelligentTieringConfiguration
- PutBucketIntelligentTieringConfiguration
- GetBucketIntelligentTieringConfiguration

Request Syntax

GET /?intelligent-tiering&continuation-token=ContinuationToken HTTP/1.1
Host: Bucket.s3.amazonaws.com

URI Request Parameters

The request uses the following URI parameters.

Bucket (p. 200)

The name of the Amazon S3 bucket whose configuration you want to modify or retrieve.

Required: Yes

continuation-token (p. 200)

The ContinuationToken that represents a placeholder from where this request should begin.

Request Body

The request does not have a request body.

Response Syntax

HTTP/1.1 200
<?xml version="1.0" encoding="UTF-8"?>
<ListBucketIntelligentTieringConfigurationsOutput>
  <IsTruncated>boolean</IsTruncated>
  <ContinuationToken>string</ContinuationToken>
  <NextContinuationToken>string</NextContinuationToken>
  <IntelligentTieringConfiguration>
Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in XML format by the service.

ListBucketIntelligentTieringConfigurationsOutput (p. 200)

Root level tag for the ListBucketIntelligentTieringConfigurationsOutput parameters.

Required: Yes

ContinuationToken (p. 200)

The ContinuationToken that represents a placeholder from where this request should begin.

Type: String

IntelligentTieringConfiguration (p. 200)

The list of S3 Intelligent-Tiering configurations for a bucket.

Type: Array of IntelligentTieringConfiguration (p. 573) data types

IsTruncated (p. 200)

Indicates whether the returned list of analytics configurations is complete. A value of true indicates that the list is not complete and the NextContinuationToken will be provided for a subsequent request.

Type: Boolean

NextContinuationToken (p. 200)

The marker used to continue this inventory configuration listing. Use the NextContinuationToken from this response to continue the listing in a subsequent request. The continuation token is an opaque value that Amazon S3 understands.

Type: String
See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3
ListBucketInventoryConfigurations
Service: Amazon Simple Storage Service

Returns a list of inventory configurations for the bucket. You can have up to 1,000 analytics configurations per bucket.

This operation supports list pagination and does not return more than 100 configurations at a time. Always check the IsTruncated element in the response. If there are no more configurations to list, IsTruncated is set to false. If there are more configurations to list, IsTruncated is set to true, and there is a value in NextContinuationToken. You use the NextContinuationToken value to continue the pagination of the list by passing the value in continuation-token in the request to GET the next page.

To use this operation, you must have permissions to perform the s3:GetInventoryConfiguration action. The bucket owner has this permission by default. The bucket owner can grant this permission to others. For more information about permissions, see Permissions Related to Bucket Subresource Operations and Managing Access Permissions to Your Amazon S3 Resources.

For information about the Amazon S3 inventory feature, see Amazon S3 Inventory

The following operations are related to ListBucketInventoryConfigurations:

- GetBucketInventoryConfiguration
- DeleteBucketInventoryConfiguration
- PutBucketInventoryConfiguration

**Request Syntax**

```
GET /?inventory&continuation-token=ContinuationToken HTTP/1.1
Host: Bucket.s3.amazonaws.com
x-amz-expected-bucket-owner: ExpectedBucketOwner
```

**URI Request Parameters**

The request uses the following URI parameters.

**Bucket (p. 203)**

The name of the bucket containing the inventory configurations to retrieve.

Required: Yes

**continuation-token (p. 203)**

The marker used to continue an inventory configuration listing that has been truncated. Use the NextContinuationToken from a previously truncated list response to continue the listing. The continuation token is an opaque value that Amazon S3 understands.

**x-amz-expected-bucket-owner (p. 203)**

The account id of the expected bucket owner. If the bucket is owned by a different account, the request will fail with an HTTP 403 (Access Denied) error.

**Request Body**

The request does not have a request body.

**Response Syntax**

```
HTTP/1.1 200
```
<ListInventoryConfigurationsResult>
  <ContinuationToken>string</ContinuationToken>
  <InventoryConfiguration>
    <Destination>
      <S3BucketDestination>
        <AccountId>string</AccountId>
        <Bucket>string</Bucket>
        <Encryption>
          <SSE-KMS>
            <KeyId>string</KeyId>
          </SSE-KMS>
          <SSE-S3/>
        </Encryption>
        <Format>string</Format>
        <Prefix>string</Prefix>
      </S3BucketDestination>
    </Destination>
    <Filter>
      <Prefix>string</Prefix>
    </Filter>
    <Id>string</Id>
    <IncludedObjectVersions>string</IncludedObjectVersions>
    <IsEnabled>boolean</IsEnabled>
    <OptionalFields>
      <Field>string</Field>
    </OptionalFields>
    <Schedule>
      <Frequency>string</Frequency>
    </Schedule>
  </InventoryConfiguration>
  ...
</ListInventoryConfigurationsResult>

Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in XML format by the service.

ListInventoryConfigurationsResult (p. 203)

Root level tag for the ListInventoryConfigurationsResult parameters.

Required: Yes

ContinuationToken (p. 203)

If sent in the request, the marker that is used as a starting point for this inventory configuration list response.

Type: String

InventoryConfiguration (p. 203)

The list of inventory configurations for a bucket.

Type: Array of InventoryConfiguration (p. 575) data types

IsTruncated (p. 203)

Tells whether the returned list of inventory configurations is complete. A value of true indicates that the list is not complete and the NextContinuationToken is provided for a subsequent request.
Type: Boolean

NextContinuationToken (p. 203)

The marker used to continue this inventory configuration listing. Use the NextContinuationToken from this response to continue the listing in a subsequent request. The continuation token is an opaque value that Amazon S3 understands.

Type: String

Examples

Sample Request

The following request returns the inventory configurations in example-bucket.

GET /?inventory HTTP/1.1
Host: example-bucket.s3.<Region>.amazonaws.com
x-amz-date: 20160430T233541Z
Authorization: authorization string
Content-Type: text/plain

Sample Response

Delete the metric configuration with a specified ID, which disables the CloudWatch metrics with the ExampleMetrics value for the FilterId dimension.

HTTP/1.1 200 OK
x-amz-id-2: gyB+3jRPnrkN98ZajxHXR3u7EFM67bNgSAXexeEHndCX/7GRnfTXXeKUQF28IfP
x-amz-request-id: 3B3C7C725673C630
Date: Sat, 30 Apr 2016 23:29:37 GMT
Content-Type: application/xml
Content-Length: length
Connection: close
Server: AmazonS3

<?xml version="1.0" encoding="UTF-8"?>
<ListInventoryConfigurationsResult xmlns="http://s3.amazonaws.com/doc/2006-03-01/">
  <InventoryConfiguration>
    <Id>report1</Id>
    <IsEnabled>true</IsEnabled>
    <Destination>
      <S3BucketDestination>
        <Format>CSV</Format>
        <AccountId>123456789012</AccountId>
        <Bucket>arn:aws:s3:::destination-bucket</Bucket>
        <Prefix>prefix1</Prefix>
      </S3BucketDestination>
    </Destination>
    <Schedule>
      <Frequency>Daily</Frequency>
    </Schedule>
    <Filter>
      <Prefix>prefix/One</Prefix>
    </Filter>
    <IncludedObjectVersions>All</IncludedObjectVersions>
    <OptionalFields>
      <Field>Size</Field>
      <Field>LastModifiedDate</Field>
    </OptionalFields>
  </InventoryConfiguration>
</ListInventoryConfigurationsResult>
<Field>ETag</Field>
<Field>StorageClass</Field>
<Field>IsMultipartUploaded</Field>
<Field>ReplicationStatus</Field>
</OptionalFields>
</InventoryConfiguration>
<InventoryConfiguration>
<Id>report2</Id>
<IsEnabled>true</IsEnabled>
<Destination>
<S3BucketDestination>
<Format>CSV</Format>
<AccountId>123456789012</AccountId>
<Bucket>arn:aws:s3:::bucket2</Bucket>
<Prefix>prefix2</Prefix>
</S3BucketDestination>
</Destination>
<Schedule>
<Frequency>Daily</Frequency>
</Schedule>
<Filter>
<Prefix>prefix/Two</Prefix>
</Filter>
<IncludedObjectVersions>All</IncludedObjectVersions>
<OptionalFields>
<Field>Size</Field>
<Field>LastModifiedDate</Field>
<Field>ETag</Field>
<Field>StorageClass</Field>
<Field>IsMultipartUploaded</Field>
<Field>ReplicationStatus</Field>
<Field>ObjectLockRetainUntilDate</Field>
<Field>ObjectLockMode</Field>
<Field>ObjectLockLegalHoldStatus</Field>
</OptionalFields>
</InventoryConfiguration>
<InventoryConfiguration>
<Id>report3</Id>
<IsEnabled>true</IsEnabled>
<Destination>
<S3BucketDestination>
<Format>CSV</Format>
<AccountId>123456789012</AccountId>
<Bucket>arn:aws:s3:::bucket3</Bucket>
<Prefix>prefix3</Prefix>
</S3BucketDestination>
</Destination>
<Schedule>
<Frequency>Daily</Frequency>
</Schedule>
<Filter>
<Prefix>prefix/Three</Prefix>
</Filter>
<IncludedObjectVersions>All</IncludedObjectVersions>
<OptionalFields>
<Field>Size</Field>
<Field>LastModifiedDate</Field>
<Field>ETag</Field>
<Field>StorageClass</Field>
<Field>IsMultipartUploaded</Field>
<Field>ReplicationStatus</Field>
</OptionalFields>
</InventoryConfiguration>
...
<IsTruncated>false</IsTruncated>
<!-- If ContinuationToken was provided in the request. -->
See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3
ListBucketMetricsConfigurations
Service: Amazon Simple Storage Service

Lists the metrics configurations for the bucket. The metrics configurations are only for the request metrics of the bucket and do not provide information on daily storage metrics. You can have up to 1,000 configurations per bucket.

This operation supports list pagination and does not return more than 100 configurations at a time. Always check the IsTruncated element in the response. If there are no more configurations to list, IsTruncated is set to false. If there are more configurations to list, IsTruncated is set to true, and there is a value in NextContinuationToken. You use the NextContinuationToken value to continue the pagination of the list by passing the value in continuation-token in the request to GET the next page.

To use this operation, you must have permissions to perform the s3:GetMetricsConfiguration action. The bucket owner has this permission by default. The bucket owner can grant this permission to others. For more information about permissions, see Permissions Related to Bucket Subresource Operations and Managing Access Permissions to Your Amazon S3 Resources.

For more information about metrics configurations and CloudWatch request metrics, see Monitoring Metrics with Amazon CloudWatch.

The following operations are related to ListBucketMetricsConfigurations:

- PutBucketMetricsConfiguration
- GetBucketMetricsConfiguration
- DeleteBucketMetricsConfiguration

Request Syntax

GET /?metrics&continuation-token=ContinuationToken  HTTP/1.1
Host: Bucket.s3.amazonaws.com
x-amz-expected-bucket-owner: ExpectedBucketOwner

URI Request Parameters

The request uses the following URI parameters.

**Bucket** *(p. 208)*

The name of the bucket containing the metrics configurations to retrieve.

Required: Yes

**continuation-token** *(p. 208)*

The marker that is used to continue a metrics configuration listing that has been truncated. Use the NextContinuationToken from a previously truncated list response to continue the listing. The continuation token is an opaque value that Amazon S3 understands.

**x-amz-expected-bucket-owner** *(p. 208)*

The account id of the expected bucket owner. If the bucket is owned by a different account, the request will fail with an HTTP 403 (Access Denied) error.

Request Body

The request does not have a request body.
Response Syntax

```xml
HTTP/1.1 200
<?xml version="1.0" encoding="UTF-8"?>
<ListMetricsConfigurationsResult>
  <IsTruncated>boolean</IsTruncated>
  <ContinuationToken>string</ContinuationToken>
  <NextContinuationToken>string</NextContinuationToken>
  <MetricsConfiguration>
    <Filter>
      <And>
        <Prefix>string</Prefix>
        <Tag>
          <Key>string</Key>
          <Value>string</Value>
        </Tag>
        ...
      </And>
    </Filter>
    <Id>string</Id>
  </MetricsConfiguration>
  ...
</ListMetricsConfigurationsResult>
```

Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in XML format by the service.

**ListMetricsConfigurationsResult (p. 209)**

Root level tag for the ListMetricsConfigurationsResult parameters.

Required: Yes

**ContinuationToken (p. 209)**

The marker that is used as a starting point for this metrics configuration list response. This value is present if it was sent in the request.

Type: String

**IsTruncated (p. 209)**

Indicates whether the returned list of metrics configurations is complete. A value of true indicates that the list is not complete and the NextContinuationToken will be provided for a subsequent request.

Type: Boolean

**MetricsConfiguration (p. 209)**

The list of metrics configurations for a bucket.

Type: Array of MetricsConfiguration (p. 597) data types
NextContinuationToken (p. 209)

The marker used to continue a metrics configuration listing that has been truncated. Use the NextContinuationToken from a previously truncated list response to continue the listing. The continuation token is an opaque value that Amazon S3 understands.

Type: String

Examples

Sample Request

Delete the metric configuration with a specified ID, which disables the CloudWatch metrics with the ExampleMetrics value for the FilterId dimension.

GET /?metrics HTTP/1.1
Host: examplebucket.s3.<Region>.amazonaws.com
x-amz-date: Thu, 15 Nov 2016 00:17:21 GMT
Authorization: signatureValue

Sample Response

Delete the metric configuration with a specified ID, which disables the CloudWatch metrics with the ExampleMetrics value for the FilterId dimension.

HTTP/1.1 200 OK
x-amz-id-2: ITnGT1y4REXAMPLEPi4hklTXouTfohcUjo0iCPEXAMPLEEutBj3M7fPGLW02SEWp
x-amz-request-id: 51991EXAMPLE5321
Date: Thu, 15 Nov 2016 00:17:22 GMT
Server: AmazonS3
Content-Length: 758

<?xml version="1.0" encoding="UTF-8"?>
  <MetricsConfiguration>
    <Id>EntireBucket</Id>
  </MetricsConfiguration>

  <MetricsConfiguration>
    <Id>Documents</Id>
    <Filter>
      <Prefix>documents/</Prefix>
    </Filter>
  </MetricsConfiguration>

  <MetricsConfiguration>
    <Id>BlueDocuments</Id>
    <Filter>
      <And>
        <Prefix>documents/</Prefix>
        <Tag>
          <Key>class</Key>
          <Value>blue</Value>
        </Tag>
      </And>
      
    </Filter>
  </MetricsConfiguration>

  <IsTruncated>false</IsTruncated>
</ListMetricsConfigurationsResult>
See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3
ListBuckets
Service: Amazon Simple Storage Service

Returns a list of all buckets owned by the authenticated sender of the request.

Request Syntax

GET / HTTP/1.1

URI Request Parameters

The request does not use any URI parameters.

Request Body

The request does not have a request body.

Response Syntax

HTTP/1.1 200
<?xml version="1.0" encoding="UTF-8"?>
<ListAllMyBucketsResult>
  <Buckets>
    <Bucket>
      <CreationDate>timestamp</CreationDate>
      <Name>string</Name>
    </Bucket>
  </Buckets>
  <Owner>
    <DisplayName>string</DisplayName>
    <ID>string</ID>
  </Owner>
</ListAllMyBucketsResult>

Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in XML format by the service.

ListAllMyBucketsResult (p. 212)

  Root level tag for the ListAllMyBucketsResult parameters.

  Required: Yes

Buckets (p. 212)

  The list of buckets owned by the requestor.

  Type: Array of Bucket (p. 523) data types

Owner (p. 212)

  The owner of the buckets listed.

  Type: Owner (p. 617) data type

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:
• AWS Command Line Interface
• AWS SDK for .NET
• AWS SDK for C++
• AWS SDK for Go
• AWS SDK for Java
• AWS SDK for JavaScript
• AWS SDK for PHP V3
• AWS SDK for Python
• AWS SDK for Ruby V3
ListMultipartUploads
Service: Amazon Simple Storage Service

This operation lists in-progress multipart uploads. An in-progress multipart upload is a multipart upload that has been initiated using the Initiate Multipart Upload request, but has not yet been completed or aborted.

This operation returns at most 1,000 multipart uploads in the response. 1,000 multipart uploads is the maximum number of uploads a response can include, which is also the default value. You can further limit the number of uploads in a response by specifying the max-uploads parameter in the response. If additional multipart uploads satisfy the list criteria, the response will contain an IsTruncated element with the value true. To list the additional multipart uploads, use the key-marker and upload-id-marker request parameters.

In the response, the uploads are sorted by key. If your application has initiated more than one multipart upload using the same object key, then uploads in the response are first sorted by key. Additionally, uploads are sorted in ascending order within each key by the upload initiation time.

For more information on multipart uploads, see Uploading Objects Using Multipart Upload.

For information on permissions required to use the multipart upload API, see Multipart Upload API and Permissions.

The following operations are related to ListMultipartUploads:
- CreateMultipartUpload
- UploadPart
- CompleteMultipartUpload
- ListParts
- AbortMultipartUpload

Request Syntax

GET /?uploads&delimiter=Delimiter&encoding-type=EncodingType&key-marker=KeyMarker&max-uploads=MaxUploads&prefix=Prefix&upload-id-marker=UploadIdMarker HTTP/1.1
Host: Bucket.s3.amazonaws.com
x-amz-expected-bucket-owner: ExpectedBucketOwner

URI Request Parameters

The request uses the following URI parameters.

Bucket (p. 214)

The name of the bucket to which the multipart upload was initiated.

When using this API with an access point, you must direct requests to the access point hostname. The access point hostname takes the form AccessPointName-AccountId.s3-accesspoint.Region.amazonaws.com. When using this operation with an access point through the AWS SDKs, you provide the access point ARN in place of the bucket name. For more information about access point ARNs, see Using Access Points in the Amazon Simple Storage Service Developer Guide.

When using this API with Amazon S3 on Outposts, you must direct requests to the S3 on Outposts hostname. The S3 on Outposts hostname takes the form AccessPointName-AccountId.outpostId.s3-outposts.Region.amazonaws.com. When using this operation using S3 on Outposts through the
AWS SDKs, you provide the Outposts bucket ARN in place of the bucket name. For more information about S3 on Outposts ARNs, see Using S3 on Outposts in the Amazon Simple Storage Service Developer Guide.

Required: Yes

delimiter (p. 214)

Character you use to group keys.

All keys that contain the same string between the prefix, if specified, and the first occurrence of the delimiter after the prefix are grouped under a single result element, CommonPrefixes. If you don't specify the prefix parameter, then the substring starts at the beginning of the key. The keys that are grouped under CommonPrefixes result element are not returned elsewhere in the response.

encoding-type (p. 214)

Requests Amazon S3 to encode the object keys in the response and specifies the encoding method to use. An object key may contain any Unicode character; however, XML 1.0 parser cannot parse some characters, such as characters with an ASCII value from 0 to 10. For characters that are not supported in XML 1.0, you can add this parameter to request that Amazon S3 encode the keys in the response.

Valid Values: url

key-marker (p. 214)

Together with upload-id-marker, this parameter specifies the multipart upload after which listing should begin.

If upload-id-marker is not specified, only the keys lexicographically greater than the specified key-marker will be included in the list.

If upload-id-marker is specified, any multipart uploads for a key equal to the key-marker might also be included, provided those multipart uploads have upload IDs lexicographically greater than the specified upload-id-marker.

max-uploads (p. 214)

Sets the maximum number of multipart uploads, from 1 to 1,000, to return in the response body. 1,000 is the maximum number of uploads that can be returned in a response.

prefix (p. 214)

Lists in-progress uploads only for those keys that begin with the specified prefix. You can use prefixes to separate a bucket into different grouping of keys. (You can think of using prefix to make groups in the same way you'd use a folder in a file system.)

upload-id-marker (p. 214)

Together with key-marker, specifies the multipart upload after which listing should begin. If key-marker is not specified, the upload-id-marker parameter is ignored. Otherwise, any multipart uploads for a key equal to the key-marker might be included in the list only if they have an upload ID lexicographically greater than the specified upload-id-marker.

x-amz-expected-bucket-owner (p. 214)

The account id of the expected bucket owner. If the bucket is owned by a different account, the request will fail with an HTTP 403 (Access Denied) error.

Request Body

The request does not have a request body.
Response Syntax

```
HTTP/1.1 200
<?xml version="1.0" encoding="UTF-8"?>
<ListMultipartUploadsResult>
  <Bucket>string</Bucket>
  <KeyMarker>string</KeyMarker>
  <UploadIdMarker>string</UploadIdMarker>
  <NextKeyMarker>string</NextKeyMarker>
  <Prefix>string</Prefix>
  <Delimiter>string</Delimiter>
  <NextUploadIdMarker>string</NextUploadIdMarker>
  <MaxUploads>integer</MaxUploads>
  <IsTruncated>boolean</IsTruncated>
  <Upload>
    <Initiated>timestamp</Initiated>
    <Initiator>
      <DisplayName>string</DisplayName>
      <ID>string</ID>
    </Initiator>
    <Key>string</Key>
    <Owner>
      <DisplayName>string</DisplayName>
      <ID>string</ID>
    </Owner>
    <StorageClass>string</StorageClass>
    <UploadId>string</UploadId>
  </Upload>
  ...<CommonPrefixes>
    <Prefix>string</Prefix>
  </CommonPrefixes>
  ...<EncodingType>string</EncodingType>
</ListMultipartUploadsResult>
```

Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in XML format by the service.

**ListMultipartUploadsResult (p. 216)**

Root level tag for the ListMultipartUploadsResult parameters.

Required: Yes

**Bucket (p. 216)**

The name of the bucket to which the multipart upload was initiated.

Type: String

**CommonPrefixes (p. 216)**

If you specify a delimiter in the request, then the result returns each distinct key prefix containing the delimiter in a CommonPrefixes element. The distinct key prefixes are returned in the Prefix child element.

Type: Array of CommonPrefix (p. 528) data types

**Delimiter (p. 216)**

Contains the delimiter you specified in the request. If you don't specify a delimiter in your request, this element is absent from the response.
Type: String

**EncodingType (p. 216)**

Encoding type used by Amazon S3 to encode object keys in the response.

If you specify encoding-type request parameter, Amazon S3 includes this element in the response, and returns encoded key name values in the following response elements:

- **Delimiter**
- **KeyMarker**
- **Prefix**
- **NextKeyMarker**
- **Key**

Type: String

Valid Values: url

**IsTruncated (p. 216)**

Indicates whether the returned list of multipart uploads is truncated. A value of true indicates that the list was truncated. The list can be truncated if the number of multipart uploads exceeds the limit allowed or specified by max uploads.

Type: Boolean

**KeyMarker (p. 216)**

The key at or after which the listing began.

Type: String

**MaxUploads (p. 216)**

Maximum number of multipart uploads that could have been included in the response.

Type: Integer

**NextKeyMarker (p. 216)**

When a list is truncated, this element specifies the value that should be used for the key-marker request parameter in a subsequent request.

Type: String

**NextUploadIdMarker (p. 216)**

When a list is truncated, this element specifies the value that should be used for the upload-id-marker request parameter in a subsequent request.

Type: String

**Prefix (p. 216)**

When a prefix is provided in the request, this field contains the specified prefix. The result contains only keys starting with the specified prefix.

Type: String

**Upload (p. 216)**

Container for elements related to a particular multipart upload. A response can contain zero or more Upload elements.

Type: Array of MultipartUpload (p. 599) data types

**UploadIdMarker (p. 216)**

Upload ID after which listing began.
Type: String

Examples

Sample Request

The following request lists three multipart uploads. The request specifies the `max-uploads` request parameter to set the maximum number of multipart uploads to return in the response body.

```
GET /?uploads&max-uploads=3 HTTP/1.1
Host: example-bucket.s3.<Region>.amazonaws.com
Date: Mon, 1 Nov 2010 20:34:56 GMT
Authorization: authorization string
```

Sample Response

The following sample response indicates that the multipart upload list was truncated and provides the `NextKeyMarker` and the `NextUploadIdMarker` elements. You specify these values in your subsequent requests to read the next set of multipart uploads. That is, send a subsequent request specifying `key-marker=my-movie2.m2ts` (value of the `NextKeyMarker` element) and `upload-id-marker=YW55IGlkZWEgd2h5IGVsdmluZydzIHVwbG9hZCBmYWJsZWQ` (value of the `NextUploadIdMarker`).

The sample response also shows a case of two multipart uploads in progress with the same key (`my-movie.m2ts`). That is, the response shows two uploads with the same key. This response shows the uploads sorted by key, and within each key the uploads are sorted in ascending order by the time the multipart upload was initiated.

```
<?xml version="1.0" encoding="UTF-8"?>
  <Bucket>bucket</Bucket>
  <KeyMarker></KeyMarker>
  <UploadIdMarker></UploadIdMarker>
  <NextKeyMarker>my-movie.m2ts</NextKeyMarker>
  <NextUploadIdMarker>YW55IGlkZWEgd2h5IGVsdmluZydzIHVwbG9hZCBmYWJsZWQ</NextUploadIdMarker>
  <MaxUploads>3</MaxUploads>
  <IsTruncated>true</IsTruncated>
  <Upload>
    <Key>my-divisor</Key>
    <UploadId>XMgbGlrZSBibHNpbnRvcnkgYm9yZWFsIGF2aWYgdHJhY2tncm91bmQgaHR0cDovL3d3dy5vZ2dvdXJ6LmVuYWJsZWQ=
      <Initiator>
        <ID>arn:aws:iam::111122223333:user/user1-11111a31-17b5-4fb7-9df5-b111111f13de</ID>
        <DisplayName>user1-11111a31-17b5-4fb7-9df5-b111111f13de</DisplayName>
      </Initiator>
      <Owner>
        <ID>75aa57f09aa0c8caebaf8c24e9d10f8e7f6c078efc7c6caea54ba06a</ID>
        <DisplayName>OwnerDisplayName</DisplayName>
      </Owner>
      <StorageClass>STANDARD</StorageClass>
    </UploadId>
  </Upload>
</ListMultipartUploadsResult>
```
Sample Request: Using the delimiter and the prefix parameters

Assume you have a multipart upload in progress for the following keys in your bucket, example-bucket.

- photos/2006/January/sample.jpg
- photos/2006/February/sample.jpg
- photos/2006/March/sample.jpg
- videos/2006/March/sample.wmv
- sample.jpg

The following list multipart upload request specifies the delimiter parameter with value "/".

```
GET /?uploads&delimiter=/ HTTP/1.1
Host: example-bucket.s3.<Region>.amazonaws.com
Date: Mon, 1 Nov 2010 20:34:56 GMT
Authorization: authorization string
```

Sample Response

The following sample response lists multipart uploads on the specified bucket, example-bucket.

The response returns multipart upload for the sample.jpg key in an <Upload> element.

However, because all the other keys contain the specified delimiter, a distinct substring, from the beginning of the key to the first occurrence of the delimiter, from each of these keys is returned in a
<CommonPrefixes> element. The key substrings, photos/ and videos/ in the <CommonPrefixes> element, indicate that there are one or more in-progress multipart uploads with these key prefixes.

This is a useful scenario if you use key prefixes for your objects to create a logical folder like structure. In this case, you can interpret the result as the folders photos/ and videos/ have one or more multipart uploads in progress.

Sample Request

In addition to the delimiter parameter, you can filter results by adding a prefix parameter as shown in the following request.

```
GET /?uploads&delimiter=/&prefix=photos/2006/ HTTP/1.1
Host: example-bucket.s3.<Region>.amazonaws.com
Date: Mon, 1 Nov 2010 20:34:56 GMT
Authorization: authorization string
```

Sample Response

In this case, the response will include only multipart uploads for keys that start with the specified prefix. The value returned in the <CommonPrefixes> element is a substring from the beginning of the key to the first occurrence of the specified delimiter after the prefix.
<?xml version="1.0" encoding="UTF-8"?>
  <Bucket>example-bucket</Bucket>
  <KeyMarker/>
  <UploadIdMarker/>
  <NextKeyMarker/>
  <NextUploadIdMarker/>
  <Delimiter>/</Delimiter>
  <Prefix>photos/2006/</Prefix>
  <MaxUploads>1000</MaxUploads>
  <IsTruncated>false</IsTruncated>
  <CommonPrefixes>
    <Prefix>photos/2006/February/</Prefix>
  </CommonPrefixes>
  <CommonPrefixes>
    <Prefix>photos/2006/January/</Prefix>
  </CommonPrefixes>
  <CommonPrefixes>
    <Prefix>photos/2006/March/</Prefix>
  </CommonPrefixes>
</ListMultipartUploadsResult>

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3
ListObjects
Service: Amazon Simple Storage Service

Returns some or all (up to 1,000) of the objects in a bucket. You can use the request parameters as selection criteria to return a subset of the objects in a bucket. A 200 OK response can contain valid or invalid XML. Be sure to design your application to parse the contents of the response and handle it appropriately.

Important
This API has been revised. We recommend that you use the newer version, ListObjectsV2, when developing applications. For backward compatibility, Amazon S3 continues to support ListObjects.

The following operations are related to ListObjects:

- ListObjectsV2
- GetObject
- PutObject
- CreateBucket
- ListBuckets

Request Syntax

GET /?delimiter=Delimiter&encoding-type=EncodingType&marker=Marker&max-keys=MaxKeys&prefix=Prefix HTTP/1.1
Host: Bucket.s3.amazonaws.com
x-amz-request-payer: RequestPayer
x-amz-expected-bucket-owner: ExpectedBucketOwner

URI Request Parameters

The request uses the following URI parameters.

Bucket (p. 222)

The name of the bucket containing the objects.

When using this API with an access point, you must direct requests to the access point hostname. The access point hostname takes the form AccessPointName-AccountId.s3-accesspoint.Region.amazonaws.com. When using this operation with an access point through the AWS SDKs, you provide the access point ARN in place of the bucket name. For more information about access point ARNs, see Using Access Points in the Amazon Simple Storage Service Developer Guide.

When using this API with Amazon S3 on Outposts, you must direct requests to the S3 on Outposts hostname. The S3 on Outposts hostname takes the form AccessPointName-AccountId.outpostId.s3-outposts.Region.amazonaws.com. When using this operation using S3 on Outposts through the AWS SDKs, you provide the Outposts bucket ARN in place of the bucket name. For more information about S3 on Outposts ARNs, see Using S3 on Outposts in the Amazon Simple Storage Service Developer Guide.

Required: Yes

delimiter (p. 222)

A delimiter is a character you use to group keys.
encoding-type (p. 222)

Requests Amazon S3 to encode the object keys in the response and specifies the encoding method to use. An object key may contain any Unicode character; however, XML 1.0 parser cannot parse some characters, such as characters with an ASCII value from 0 to 10. For characters that are not supported in XML 1.0, you can add this parameter to request that Amazon S3 encode the keys in the response.

Valid Values: url

marker (p. 222)

Specifies the key to start with when listing objects in a bucket.

max-keys (p. 222)

Sets the maximum number of keys returned in the response. By default the API returns up to 1,000 key names. The response might contain fewer keys but will never contain more.

prefix (p. 222)

Limits the response to keys that begin with the specified prefix.

x-amz-expected-bucket-owner (p. 222)

The account id of the expected bucket owner. If the bucket is owned by a different account, the request will fail with an HTTP 403 (Access Denied) error.

x-amz-request-payer (p. 222)

Confirms that the requester knows that she or he will be charged for the list objects request. Bucket owners need not specify this parameter in their requests.

Valid Values: requester

Request Body

The request does not have a request body.

Response Syntax

```
HTTP/1.1 200
<?xml version="1.0" encoding="UTF-8"?>
<ListBucketResult>
  <IsTruncated>boolean</IsTruncated>
  <Marker>string</Marker>
  <NextMarker>string</NextMarker>
  <Contents>
    <ETag>string</ETag>
    <Key>string</Key>
    <LastModified>timestamp</LastModified>
    <Owner>
      <DisplayName>string</DisplayName>
      <ID>string</ID>
    </Owner>
    <Size>integer</Size>
    <StorageClass>string</StorageClass>
  </Contents>
  ...
  <Name>string</Name>
  <Prefix>string</Prefix>
  <Delimiter>string</Delimiter>
  <MaxKeys>integer</MaxKeys>
  <CommonPrefixes>
```

API Version 2006-03-01
223
Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in XML format by the service.

**ListBucketResult (p. 223)**

Root level tag for the ListBucketResult parameters.

Required: Yes

**CommonPrefixes (p. 223)**

- All of the keys rolled up in a common prefix count as a single return when calculating the number of returns.

- A response can contain CommonPrefixes only if you specify a delimiter.

- CommonPrefixes contains all (if there are any) keys between Prefix and the next occurrence of the string specified by the delimiter.

- CommonPrefixes lists keys that act like subdirectories in the directory specified by Prefix.

  For example, if the prefix is notes/ and the delimiter is a slash (/) as in notes/summer/july, the common prefix is notes/summer/. All of the keys that roll up into a common prefix count as a single return when calculating the number of returns.

  Type: Array of CommonPrefix (p. 528) data types

**Contents (p. 223)**

- Metadata about each object returned.

  Type: Array of Object (p. 606) data types

**Delimiter (p. 223)**

- Causes keys that contain the same string between the prefix and the first occurrence of the delimiter to be rolled up into a single result element in the CommonPrefixes collection. These rolled-up keys are not returned elsewhere in the response. Each rolled-up result counts as only one return against the MaxKeys value.

  Type: String

**EncodingType (p. 223)**

- Encoding type used by Amazon S3 to encode object keys in the response.

  Type: String

  Valid Values: url

**IsTruncated (p. 223)**

- A flag that indicates whether Amazon S3 returned all of the results that satisfied the search criteria.

  Type: Boolean
Marker (p. 223)

Indicates where in the bucket listing begins. Marker is included in the response if it was sent with the request.

Type: String

MaxKeys (p. 223)

The maximum number of keys returned in the response body.

Type: Integer

Name (p. 223)

The bucket name.

Type: String

NextMarker (p. 223)

When response is truncated (the IsTruncated element value in the response is true), you can use the key name in this field as marker in the subsequent request to get next set of objects. Amazon S3 lists objects in alphabetical order Note: This element is returned only if you have delimiter request parameter specified. If response does not include the NextMarker and it is truncated, you can use the value of the last Key in the response as the marker in the subsequent request to get the next set of object keys.

Type: String

Prefix (p. 223)

Keys that begin with the indicated prefix.

Type: String

The specified bucket does not exist.

Examples

Sample Request

This request returns the objects in BucketName.

GET / HTTP/1.1
Host: BucketName.s3.<Region>.amazonaws.com
Date: Wed, 12 Oct 2009 17:50:00 GMT
Authorization: authorization string
Content-Type: text/plain

Sample Response

This example illustrates one usage of ListObjects.

<?xml version="1.0" encoding="UTF-8"?>
  <Name>bucket</Name>
  <Prefix/>
  <Marker/>
  <MaxKeys>1000</MaxKeys>
Sample Request: Using request parameters

This example lists up to 40 keys in the quotes bucket that start with N and occur lexicographically after Ned.

GET /?prefix=N&marker=Ned&max-keys=40 HTTP/1.1
Host: quotes.s3.<Region>.amazonaws.com
Date: Wed, 01 Mar 2006 12:00:00 GMT
Authorization: authorization string

Sample Response

This example illustrates one usage of ListObjects.

HTTP/1.1 200 OK
x-amz-id-2: gyB+3jRPnrkN9H6ZajxH3r37EFM67bNgSAXeexeEHnmCX/7GRnTXXeRUQpF28IfP
x-amz-request-id: 3B3C7C73567363C630
Date: Wed, 01 Mar 2006 12:00:00 GMT
Content-Type: application/xml
Content-Length: 302
Connection: close
Server: AmazonS3

<?xml version="1.0" encoding="UTF-8"?>
  <Name>quotes</Name>
  <Prefix>N</Prefix>
  <Marker>Ned</Marker>
  <MaxKeys>40</MaxKeys>
  <IsTruncated>false</IsTruncated>
  <Contents>
    <Key>Nelson</Key>
    <LastModified>2006-01-01T12:00:00.000Z</LastModified>
  </Contents>
</ListBucketResult>
Sample Request: Using a prefix and delimiter

For this example, we assume that you have the following keys in your bucket:

- sample.jpg
- photos/2006/January/sample.jpg
- photos/2006/February/sample2.jpg
- photos/2006/February/sample3.jpg
- photos/2006/February/sample4.jpg

The following GET request specifies the delimiter parameter with value `/`.

```plaintext
GET /?delimiter=/ HTTP/1.1
Host: example-bucket.s3.<Region>.amazonaws.com
Date: Wed, 01 Mar 2006 12:00:00 GMT
Authorization: authorization string
```

Sample Response

The key sample.jpg does not contain the delimiter character, and Amazon S3 returns it in the `Contents` element in the response. However, all other keys contain the delimiter character. Amazon S3 groups these keys and returns a single `CommonPrefixes` element with prefix value `photos/` that is a substring from the beginning of these keys to the first occurrence of the specified delimiter.
Sample Request

The following GET request specifies the delimiter parameter with the value `/`, and the prefix parameter with the value `photos/2006/`.

```
GET /?prefix=photos/2006/&delimiter=/ HTTP/1.1
Host: example-bucket.s3.<Region>.amazonaws.com
Date: Wed, 01 Mar 2006 12:00:00 GMT
Authorization: authorization string
```

Sample Response

In response, Amazon S3 returns only the keys that start with the specified prefix. It uses the delimiter character to group keys that contain the same substring until the first occurrence of the delimiter character after the specified prefix. For each such key group, Amazon S3 returns one `<CommonPrefixes>` element in the response. The keys grouped under this `<CommonPrefixes>` element are not returned elsewhere in the response. The value returned in the `<CommonPrefixes>` element is a substring that starts at the beginning of the key and ends at the first occurrence of the specified delimiter after the prefix.

```
  <Name>example-bucket</Name>
  <Prefix>photos/2006/</Prefix>
  <Marker></Marker>
  <MaxKeys>1000</MaxKeys>
  <Delimiter>/</Delimiter>
  <IsTruncated>false</IsTruncated>
  <CommonPrefixes>
    <Prefix>photos/2006/February/</Prefix>
  </CommonPrefixes>
  <CommonPrefixes>
    <Prefix>photos/2006/January/</Prefix>
  </CommonPrefixes>
</ListBucketResult>
```

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:
• AWS Command Line Interface
• AWS SDK for .NET
• AWS SDK for C++
• AWS SDK for Go
• AWS SDK for Java
• AWS SDK for JavaScript
• AWS SDK for PHP V3
• AWS SDK for Python
• AWS SDK for Ruby V3
ListObjectsV2
Service: Amazon Simple Storage Service

Returns some or all (up to 1,000) of the objects in a bucket. You can use the request parameters as selection criteria to return a subset of the objects in a bucket. A 200 OK response can contain valid or invalid XML. Make sure to design your application to parse the contents of the response and handle it appropriately.

To use this operation, you must have READ access to the bucket.

To use this operation in an AWS Identity and Access Management (IAM) policy, you must have permissions to perform the s3:ListBucket action. The bucket owner has this permission by default and can grant this permission to others. For more information about permissions, see Permissions Related to Bucket Subresource Operations and Managing Access Permissions to Your Amazon S3 Resources.

Important
This section describes the latest revision of the API. We recommend that you use this revised API for application development. For backward compatibility, Amazon S3 continues to support the prior version of this API, ListObjects.

To get a list of your buckets, see ListBuckets.

The following operations are related to ListObjectsV2:

- GetObject
- PutObject
- CreateBucket

Request Syntax

GET /?list-type=2&continuation-token=ContinuationToken&delimiter=Delimiter&encoding-type=EncodingType&fetch-owner=FetchOwner&max-keys=MaxKeys&prefix=Prefix&start-after=StartAfter HTTP/1.1
Host: Bucket.s3.amazonaws.com
x-amz-request-payer: RequestPayer
x-amz-expected-bucket-owner: ExpectedBucketOwner

URI Request Parameters

The request uses the following URI parameters.

Bucket (p. 230)

Bucket name to list.

When using this API with an access point, you must direct requests to the access point hostname. The access point hostname takes the form AccessPointName-AccountId.s3-accesspoint.Region.amazonaws.com. When using this operation with an access point through the AWS SDKs, you provide the access point ARN in place of the bucket name. For more information about access point ARNs, see Using Access Points in the Amazon Simple Storage Service Developer Guide.

When using this API with Amazon S3 on Outposts, you must direct requests to the S3 on Outposts hostname. The S3 on Outposts hostname takes the form AccessPointName-AccountId.outpostId.s3-outposts.Region.amazonaws.com. When using this operation using S3 on Outposts through the AWS SDKs, you provide the Outposts bucket ARN in place of the bucket name. For more information
about S3 on Outposts ARNs, see Using S3 on Outposts in the Amazon Simple Storage Service Developer Guide.

Required: Yes

continuation-token (p. 230)

ContinuationToken indicates Amazon S3 that the list is being continued on this bucket with a token. ContinuationToken is obfuscated and is not a real key.

delimiter (p. 230)

A delimiter is a character you use to group keys.

encoding-type (p. 230)

Encoding type used by Amazon S3 to encode object keys in the response.

Valid Values: url

fetch-owner (p. 230)

The owner field is not present in listV2 by default, if you want to return owner field with each key in the result then set the fetch owner field to true.

max-keys (p. 230)

Sets the maximum number of keys returned in the response. By default the API returns up to 1,000 key names. The response might contain fewer keys but will never contain more.

prefix (p. 230)

Limits the response to keys that begin with the specified prefix.

start-after (p. 230)

StartAfter is where you want Amazon S3 to start listing from. Amazon S3 starts listing after this specified key. StartAfter can be any key in the bucket.

x-amz-expected-bucket-owner (p. 230)

The account id of the expected bucket owner. If the bucket is owned by a different account, the request will fail with an HTTP 403 (Access Denied) error.

x-amz-request-payer (p. 230)

Confirms that the requester knows that she or he will be charged for the list objects request in V2 style. Bucket owners need not specify this parameter in their requests.

Valid Values: requester

Request Body

The request does not have a request body.

Response Syntax

```xml
<?xml version="1.0" encoding="UTF-8"?>
<ListBucketResult>
  <IsTruncated>boolean</IsTruncated>
  <Contents>
    <ETag>string</ETag>
    <Key>string</Key>
  </Contents>
</ListBucketResult>
```

API Version 2006-03-01

231
Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in XML format by the service.

**ListBucketResult (p. 231)**

Root level tag for the ListBucketResult parameters.

Required: Yes

**CommonPrefixes (p. 231)**

All of the keys rolled up into a common prefix count as a single return when calculating the number of returns.

A response can contain CommonPrefixes only if you specify a delimiter.

CommonPrefixes contains all (if there are any) keys between Prefix and the next occurrence of the string specified by a delimiter.

CommonPrefixes lists keys that act like subdirectories in the directory specified by Prefix.

For example, if the prefix is notes/ and the delimiter is a slash (/) as in notes/summer/july, the common prefix is notes/summer/. All of the keys that roll up into a common prefix count as a single return when calculating the number of returns.

Type: Array of CommonPrefix (p. 528) data types

**Contents (p. 231)**

Metadata about each object returned.

Type: Array of Object (p. 606) data types

**ContinuationToken (p. 231)**

If ContinuationToken was sent with the request, it is included in the response.
Type: String

**Delimiter (p. 231)**

Causes keys that contain the same string between the prefix and the first occurrence of the delimiter to be rolled up into a single result element in the CommonPrefixes collection. These rolled-up keys are not returned elsewhere in the response. Each rolled-up result counts as only one return against the MaxKeys value.

Type: String

**EncodingType (p. 231)**

Encoding type used by Amazon S3 to encode object key names in the XML response.

If you specify the encoding-type request parameter, Amazon S3 includes this element in the response, and returns encoded key name values in the following response elements:

- Delimiter, Prefix, Key, and StartAfter.

Type: String

Valid Values: url

**IsTruncated (p. 231)**

Set to false if all of the results were returned. Set to true if more keys are available to return. If the number of results exceeds that specified by MaxKeys, all of the results might not be returned.

Type: Boolean

**KeyCount (p. 231)**

KeyCount is the number of keys returned with this request. KeyCount will always be less than equals to MaxKeys field. Say you ask for 50 keys, your result will include less than equals 50 keys

Type: Integer

**MaxKeys (p. 231)**

Sets the maximum number of keys returned in the response. By default the API returns up to 1,000 key names. The response might contain fewer keys but will never contain more.

Type: Integer

**Name (p. 231)**

The bucket name.

When using this API with an access point, you must direct requests to the access point hostname. The access point hostname takes the form AccessPointName-AccountId.s3-accesspoint.Region.amazonaws.com. When using this operation with an access point through the AWS SDKs, you provide the access point ARN in place of the bucket name. For more information about access point ARNs, see Using Access Points in the Amazon Simple Storage Service Developer Guide.

When using this API with Amazon S3 on Outposts, you must direct requests to the S3 on Outposts hostname. The S3 on Outposts hostname takes the form AccessPointName-AccountId.outpostId.s3-outposts.Region.amazonaws.com. When using this operation using S3 on Outposts through the AWS SDKs, you provide the Outposts bucket ARN in place of the bucket name. For more information about S3 on Outposts ARNs, see Using S3 on Outposts in the Amazon Simple Storage Service Developer Guide.
NextContinuationToken (p. 231)

NextContinuationToken is sent when isTruncated is true, which means there are more keys in the bucket that can be listed. The next list requests to Amazon S3 can be continued with this NextContinuationToken. NextContinuationToken is obfuscated and is not a real key.

Type: String

Prefix (p. 231)

Keys that begin with the indicated prefix.

Type: String

StartAfter (p. 231)

If StartAfter was sent with the request, it is included in the response.

Type: String

The specified bucket does not exist.

Examples

Sample Request: Listing keys

This request returns the objects in BucketName. The request specifies the list-type parameter, which indicates version 2 of the API.

```plaintext
GET /?list-type=2 HTTP/1.1
Host: bucket.s3.<Region>.amazonaws.com
x-amz-date: 20160430T233541Z
Authorization: authorization string
Content-Type: text/plain
```

Sample Response

This example illustrates one usage of ListObjectsV2.

```xml
<?xml version="1.0" encoding="UTF-8"?>
  <Name>bucket</Name>
  <Prefix/>
  <KeyCount>205</KeyCount>
  <MaxKeys>1000</MaxKeys>
  <IsTruncated>false</IsTruncated>
  <Contents>
    <Key>my-image.jpg</Key>
    <LastModified>2009-10-12T17:50:30.000Z</LastModified>
  <ETag>"fba9dede5f27731c9771645a39863328"</ETag>
  <Size>434234</Size>
  <StorageClass>STANDARD</StorageClass>
  </Contents>
  ...
</ListBucketResult>
```
Sample Request: Listing keys using the max-keys, prefix, and start-after parameters

In addition to the list-type parameter that indicates version 2 of the API, the request also specifies additional parameters to retrieve up to three keys in the quotes bucket that start with E and occur lexicographically after ExampleGuide.pdf.

```
GET /?list-type=2&max-keys=3&prefix=E&start-after=ExampleGuide.pdf HTTP/1.1
Host: quotes.s3.<Region>.amazonaws.com
x-amz-date: 20160430T232933Z
Authorization: authorization string
```

Sample Response

This example illustrates one usage of ListObjectsV2.

```
HTTP/1.1 200 OK
x-amz-id-2: gyB+3jRPnkN98ZajxHXr3u7EFM67bNgSAXexeEHndCX/7GRnfTXxReKUQF28IfP
x-amz-request-id: 3B3C7C725673C630
Date: Sat, 30 Apr 2016 23:29:37 GMT
Content-Type: application/xml
Content-Length: length
Connection: close
Server: AmazonS3

<?xml version="1.0" encoding="UTF-8"?>
  <Name>quotes</Name>
  <Prefix>E</Prefix>
  <StartAfter>ExampleGuide.pdf</StartAfter>
  <KeyCount>1</KeyCount>
  <MaxKeys>3</MaxKeys>
  <IsTruncated>false</IsTruncated>
  <Contents>
    <Key>ExampleObject.txt</Key>
    <LastModified>2013-09-17T18:07:53.000Z</LastModified>
    <ETag>"599bab3ed2c697f1d26842727561fd94"</ETag>
    <Size>857</Size>
    <StorageClass>REDUCED_REDUNDANCY</StorageClass>
  </Contents>
</ListBucketResult>
```

Sample Request: Listing keys using the prefix and delimiter parameters

This example illustrates the use of the prefix and the delimiter parameters in the request. For this example, we assume that you have the following keys in your bucket:

- sample.jpg
- photos/2006/January/sample.jpg
- photos/2006/February/sample2.jpg
- photos/2006/February/sample3.jpg
- photos/2006/February/sample4.jpg

The following GET request specifies the delimiter parameter with value /.
GET /?list-type=2&delimiter=/ HTTP/1.1
Host: example-bucket.s3.<Region>.amazonaws.com
x-amz-date: 20160430T235931Z
Authorization: authorization string

Sample Response

The key sample.jpg does not contain the delimiter character, and Amazon S3 returns it in the Contents element in the response. However, all other keys contain the delimiter character. Amazon S3 groups these keys and returns a single CommonPrefixes element with the prefix value photos/. The element is a substring that starts at the beginning of these keys and ends at the first occurrence of the specified delimiter.

Sample Request

The following request specifies the delimiter parameter with value /, and the prefix parameter with value photos/2006/.

GET /?list-type=2&prefix=photos/2006/&delimiter=/ HTTP/1.1
Host: example-bucket.s3.<Region>.amazonaws.com
x-amz-date: 20160501T000433Z
Authorization: authorization string

Sample Response

In response, Amazon S3 returns only the keys that start with the specified prefix. Further, it uses the delimiter character to group keys that contain the same substring until the first occurrence of the delimiter character after the specified prefix. For each such key group Amazon S3 returns one CommonPrefixes element in the response. The keys grouped under this CommonPrefixes element are not returned elsewhere in the response. The value returned in the CommonPrefixes element is a substring that starts at the beginning of the key and ends at the first occurrence of the specified delimiter after the prefix.
Sample Request: Using a continuation token

In this example, the initial request returns more than 1,000 keys. In response to this request, Amazon S3 returns the IsTruncated element with the value set to true and with a NextContinuationToken element.

GET /?list-type=2 HTTP/1.1
Host: bucket.s3.<Region>.amazonaws.com
Date: Mon, 02 May 2016 23:17:07 GMT
Authorization: authorization string

Sample Response: Using a continuation token

This example illustrates one usage of ListObjectsV2.

HTTP/1.1 200 OK
x-amz-id-2: gyB+3jRPnknND98ZajxHx3u7EM67bNgSAXexeEHmdCX/7GRnfoTXxReKUQF28IfP
x-amz-request-id: 3B3C7C725673C630
Date: Sat, 30 Apr 2016 23:29:37 GMT
Content-Type: application/xml
Content-Length: length
Connection: close
Server: AmazonS3

  <Name>buckett</Name>
  <Prefix></Prefix>
  <NextContinuationToken>1ueGcxLPRx1Tr/XYExHnbbYLgveDs2J/wm36Hy4vboWm=/</NextContinuationToken>
  <KeyCount>1000</KeyCount>
  <MaxKeys>1000</MaxKeys>
  <IsTruncated>true</IsTruncated>
  <Contents>
    <Key>happyface.jpg</Key>
    <LastModified>2014-11-21T19:40:05.000Z</LastModified>
  </Contents>
</ListBucketResult>
Example

In the following subsequent request, we include a continuation-token query parameter in the request with value of the &lt;NextContinuationToken&gt; from the preceding response.

```
GET /?list-type=2 HTTP/1.1
GET /?list-type=2&continuation-token=1ueGcxLPRx1Tr/XYExHnhbYlgveDs2J/wm36Hy4vbOwM= HTTP/1.1
Host: bucket.s3.<Region>.amazonaws.com
Date: Mon, 02 May 2016 23:17:07 GMT
Authorization: authorization string
```

Example

Amazon S3 returns a list of the next set of keys starting where the previous request ended.

```
HTTP/1.1 200 OK
x-amz-id-2: gyB+3jRPnrkN98ZajxHXR3u7EFM67bNgSAXexeEHmdCX/7GRnfTXxReKUQF28IfP
x-amz-request-id: 3B3C7C725673C630
Date: Sat, 30 Apr 2016 23:29:37 GMT
Content-Type: application/xml
Content-Length: length
Connection: close
Server: AmazonS3

  <Name>bucket</Name>
  <Prefix></Prefix>
  <ContinuationToken>1ueGcxLPRx1Tr/XYExHnhbYlgveDs2J/wm36Hy4vbOwM=</ContinuationToken>
  <KeyCount>112</KeyCount>
  <MaxKeys>1000</MaxKeys>
  <IsTruncated>false</IsTruncated>
  <Contents>
    <Key>happyfacex.jpg</Key>
    <LastModified>2014-11-21T19:40:05.000Z</LastModified>
    <ETag>"70ee1738b6b21e2c8a43f3a5ab0ee71"</ETag>
    <Size>1111</Size>
    <StorageClass>STANDARD</StorageClass>
  </Contents>
  ...
</ListBucketResult>
```

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
• AWS SDK for .NET
• AWS SDK for C++
• AWS SDK for Go
• AWS SDK for Java
• AWS SDK for JavaScript
• AWS SDK for PHP V3
• AWS SDK for Python
• AWS SDK for Ruby V3
**ListObjectVersions**  
*Service: Amazon Simple Storage Service*

Returns metadata about all versions of the objects in a bucket. You can also use request parameters as selection criteria to return metadata about a subset of all the object versions.

**Note**  
A 200 OK response can contain valid or invalid XML. Make sure to design your application to parse the contents of the response and handle it appropriately.

To use this operation, you must have READ access to the bucket.

This action is not supported by Amazon S3 on Outposts.

The following operations are related to **ListObjectVersions**:

- ListObjectsV2
- GetObject
- PutObject
- DeleteObject

**Request Syntax**

```
GET /?versions&delimiter=Delimiter&encoding-type=EncodingType&key-marker=KeyMarker&max-keys=MaxKeys&prefix=Prefix&version-id-marker=VersionIdMarker HTTP/1.1
Host: Bucket.s3.amazonaws.com
x-amz-expected-bucket-owner: ExpectedBucketOwner
```

**URI Request Parameters**

The request uses the following URI parameters.

**Bucket (p. 240)**

The bucket name that contains the objects.

Required: Yes

**delimiter (p. 240)**

A delimiter is a character that you specify to group keys. All keys that contain the same string between the `prefix` and the first occurrence of the delimiter are grouped under a single result element in CommonPrefixes. These groups are counted as one result against the max-keys limitation. These keys are not returned elsewhere in the response.

**encoding-type (p. 240)**

Requests Amazon S3 to encode the object keys in the response and specifies the encoding method to use. An object key may contain any Unicode character; however, XML 1.0 parser cannot parse some characters, such as characters with an ASCII value from 0 to 10. For characters that are not supported in XML 1.0, you can add this parameter to request that Amazon S3 encode the keys in the response.

Valid Values: url

**key-marker (p. 240)**

Specifies the key to start with when listing objects in a bucket.
max-keys (p. 240)

Sets the maximum number of keys returned in the response. By default the API returns up to 1,000 key names. The response might contain fewer keys but will never contain more. If additional keys satisfy the search criteria, but were not returned because max-keys was exceeded, the response contains <isTruncated>true</isTruncated>. To return the additional keys, see key-marker and version-id-marker.

prefix (p. 240)

Use this parameter to select only those keys that begin with the specified prefix. You can use prefixes to separate a bucket into different groupings of keys. (You can think of using prefix to make groups in the same way you'd use a folder in a file system.) You can use prefix with delimiter to roll up numerous objects into a single result under CommonPrefixes.

version-id-marker (p. 240)

Specifies the object version you want to start listing from.

x-amz-expected-bucket-owner (p. 240)

The account id of the expected bucket owner. If the bucket is owned by a different account, the request will fail with an HTTP 403 (Access Denied) error.

Request Body

The request does not have a request body.

Response Syntax

HTTP/1.1 200
<?xml version="1.0" encoding="UTF-8"?>
<ListVersionsResult>
  <IsTruncated>boolean</IsTruncated>
  <KeyMarker>string</KeyMarker>
  <VersionIdMarker>string</VersionIdMarker>
  <NextKeyMarker>string</NextKeyMarker>
  <NextVersionIdMarker>string</NextVersionIdMarker>
  <Version>
    <ETag>string</ETag>
    <IsLatest>boolean</IsLatest>
    <Key>string</Key>
    <LastModified>timestamp</LastModified>
    <Owner>
      <DisplayName>string</DisplayName>
      <ID>string</ID>
    </Owner>
    <Size>integer</Size>
    <StorageClass>string</StorageClass>
    <VersionId>string</VersionId>
  </Version>
  ...
  <DeleteMarker>
    <IsLatest>boolean</IsLatest>
    <Key>string</Key>
    <LastModified>timestamp</LastModified>
    <Owner>
      <DisplayName>string</DisplayName>
      <ID>string</ID>
    </Owner>
    <VersionId>string</VersionId>
  </DeleteMarker>
  ...
</ListVersionsResult>
Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in XML format by the service.

ListVersionsResult (p. 241)

Root level tag for the ListVersionsResult parameters.

Required: Yes

CommonPrefixes (p. 241)

All of the keys rolled up into a common prefix count as a single return when calculating the number of returns.

Type: Array of CommonPrefix (p. 528) data types

DeleteMarker (p. 241)

Container for an object that is a delete marker.

Type: Array of DeleteMarkerEntry (p. 545) data types

Delimiter (p. 241)

The delimiter grouping the included keys. A delimiter is a character that you specify to group keys. All keys that contain the same string between the prefix and the first occurrence of the delimiter are grouped under a single result element in CommonPrefixes. These groups are counted as one result against the max-keys limitation. These keys are not returned elsewhere in the response.

Type: String

EncodingType (p. 241)

Encoding type used by Amazon S3 to encode object key names in the XML response.

If you specify encoding-type request parameter, Amazon S3 includes this element in the response, and returns encoded key name values in the following response elements:

KeyMarker, NextKeyMarker, Prefix, Key, and Delimiter.

Type: String

Valid Values: url

IsTruncated (p. 241)

A flag that indicates whether Amazon S3 returned all of the results that satisfied the search criteria. If your results were truncated, you can make a follow-up paginated request using the NextKeyMarker.
and NextVersionIdMarker response parameters as a starting place in another request to return the rest of the results.

Type: Boolean

**KeyMarker (p. 241)**

Marks the last key returned in a truncated response.

Type: String

**MaxKeys (p. 241)**

Specifies the maximum number of objects to return.

Type: Integer

**Name (p. 241)**

The bucket name.

Type: String

**NextKeyMarker (p. 241)**

When the number of responses exceeds the value of MaxKeys, NextKeyMarker specifies the first key not returned that satisfies the search criteria. Use this value for the key-marker request parameter in a subsequent request.

Type: String

**NextVersionIdMarker (p. 241)**

When the number of responses exceeds the value of MaxKeys, NextVersionIdMarker specifies the first object version not returned that satisfies the search criteria. Use this value for the version-id-marker request parameter in a subsequent request.

Type: String

**Prefix (p. 241)**

Selects objects that start with the value supplied by this parameter.

Type: String

**Version (p. 241)**

Container for version information.

Type: Array of **ObjectVersion (p. 613)** data types

**VersionIdMarker (p. 241)**

Marks the last version of the key returned in a truncated response.

Type: String

**Examples**

**Sample Request**

The following request returns all of the versions of all of the objects in the specified bucket.
GET /?versions HTTP/1.1
Host: BucketName.s3.<Region>.amazonaws.com
Date: Wed, 28 Oct 2009 22:32:00 +0000
Authorization: authorization string (see Authenticating Requests (AWS Signature Version 4))

Sample Response

This example illustrates one usage of ListObjectVersions.

```xml
<ListVersionsResult xmlns="http://s3.amazonaws.com/doc/2006-03-01">
  <Name>bucket</Name>
  <Prefix>my</Prefix>
  <KeyMarker/>
  <VersionIdMarker/>
  <MaxKeys>5</MaxKeys>
  <IsTruncated>false</IsTruncated>
  <Version>
    <Key>my-image.jpg</Key>
    <VersionId>3/L4kqtj140Nr8X8gdrQBPdUMLjUo</VersionId>
    <LastModified>2009-10-12T17:50:30.000Z</LastModified>
    <ETag>"fba9dede5f27731c9771645a39863328"</ETag>
    <Size>434234</Size>
    <StorageClass>STANDARD</StorageClass>
    <Owner>
      <ID>75aa57f09aa0c8caeb4f8c824e99d10f8e7faeebf76c078efc7c6caea54ba06a</ID>
      <DisplayName>mtd@amazon.com</DisplayName>
    </Owner>
  </Version>
  <Version>
    <Key>my-second-image.jpg</Key>
    <VersionId>03jpff543dhffds434rfdsFDN943fdsFkdmqnh892</VersionId>
    <LastModified>2009-11-12T17:50:30.000Z</LastModified>
    <Owner>
      <ID>75aa57f09aa0c8caeb4f8c824e99d10f8e7faeebf76c078efc7c6caea54ba06a</ID>
      <DisplayName>mtd@amazon.com</DisplayName>
    </Owner>
  </Version>
  <Version>
    <Key>my-second-image.jpg</Key>
    <VersionId>QUpfdndhf8d483MNFDN93jdnJFkdmqnh893</VersionId>
    <LastModified>2009-10-10T17:50:30.000Z</LastModified>
    <ETag>"9b2cf535f27731c974343645a3985328"</ETag>
    <Size>166434</Size>
    <StorageClass>STANDARD</StorageClass>
    <Owner>
      <ID>75aa57f09aa0c8caeb4f8c824e99d10f8e7faeebf76c078efc7c6caea54ba06a</ID>
      <DisplayName>mtd@amazon.com</DisplayName>
    </Owner>
  </Version>
  <DeleteMarker>
    <Key>my-second-image.jpg</Key>
    <VersionId>03jpff543dhffds434rfdsFDN943fdsFkdmqnh892</VersionId>
    <LastModified>2009-11-12T17:50:30.000Z</LastModified>
    <Owner>
      <ID>75aa57f09aa0c8caeb4f8c824e99d10f8e7faeebf76c078efc7c6caea54ba06a</ID>
      <DisplayName>mtd@amazon.com</DisplayName>
    </Owner>
  </DeleteMarker>
  <Version>
    <Key>my-third-image.jpg</Key>
    <VersionId>03jpff543dhffds434rfdsFDN943fdsFkdmqnh892</VersionId>
    <LastModified>2009-10-15T17:50:30.000Z</LastModified>
    <Owner>
      <ID>75aa57f09aa0c8caeb4f8c824e99d10f8e7faeebf76c078efc7c6caea54ba06a</ID>
      <DisplayName>mtd@amazon.com</DisplayName>
    </Owner>
  </Version>
</ListVersionsResult>
```
Sample Request

The following request returns objects in the order they were stored, returning the most recently stored object first starting with the value for key-marker.

```
GET /?versions&key-marker=key2 HTTP/1.1
Host: s3.amazonaws.com
Pragma: no-cache
Accept: image/gif, image/x-xbitmap, image/jpeg, image/pjpeg, */*
Date: Thu, 10 Dec 2009 22:46:32 +0000
Authorization: signatureValue
```

Sample Response

This example illustrates one usage of ListObjectVersions.

```
<?xml version="1.0" encoding="UTF-8"?>
    Name=mtp-versioning-fresh</Name>
    <Prefix/>
    <KeyMarker>key2</KeyMarker>
    <VersionIdMarker/>
    <MaxKeys>1000</MaxKeys>
    <IsTruncated>false</IsTruncated>
    <Version>
        <Key>key3</Key>
        <VersionId>ISVhmK6CDDdQ5Pwfef1gcHZWmHDpcv7gfmcf29UBxsKU.</VersionId>
        <IsLatest>true</IsLatest>
        <LastModified>2009-12-09T00:19:04.000Z</LastModified>
        <ETag>"396fefef536d5ce46c7537eef978a360"</ETag>
        <Size>217</Size>
        <Owner>
            <ID>75aa57f09a0c8caeb4f8c24e99d10f8e7faeebf76c078efc7c6caea54ba06a</ID>
        </Owner>
        <StorageClass>STANDARD</StorageClass>
    </Version>
```

API Version 2006-03-01
245
<VersionId>qDhprLU80aAlCFLu2DWgXAEDgKzWarn-HS_JU0TvYqs.</VersionId>
<IsLatest>true</IsLatest>
<LastModified>2009-12-10T16:38:11.000Z</LastModified>
<Owner>
  <ID>75aa57f09aa0c8caeb4f8c2e99d10f8e7f8e9f76c078efc7c6caea54ba06a</ID>
</Owner>
</DeleteMarker>

<Version>
  <Key>sourcekey</Key>
  <VersionId>wxxQ7ezLaL5N2Si1q66Sy6x0k7uHTUbp9qiiMxNg.</VersionId>
  <IsLatest>false</IsLatest>
  <LastModified>2009-12-10T16:37:44.000Z</LastModified>
  <ETag>"396fefe5f36d5ce46c7537efc978a360"</ETag>
  <Size>217</Size>
  <Owner>
    <ID>75aa57f09aa0c8caeb4f8c2e99d10f8e7f8e9f76c078efc7c6caea54ba06a</ID>
  </Owner>
  <StorageClass>STANDARD</StorageClass>
</Version>
</ListVersionsResult>

Sample Request Using prefix

This example returns objects whose keys begin with source.

GET //?versions&prefix=source HTTP/1.1
Host: bucket.s3.<Region>.amazonaws.com
Date: Wed, 28 Oct 2009 22:32:00 +0000
Authorization: authorization string

Sample Response

This example illustrates one usage of ListObjectVersions.

<?xml version="1.0" encoding="UTF-8"?><ListVersionsResult xmlns="http://s3.amazonaws.com/doc/2006-03-01/">
  <Name>mtp-versioning-fresh</Name>
  <Prefix>source</Prefix>
  <KeyMarker/>
  <VersionIdMarker/>
  <MaxKeys>1000</MaxKeys>
  <IsTruncated>false</IsTruncated>
  <DeleteMarker>
    <Key>sourcekey</Key>
    <VersionId>qDhprLU80aAlCFLu2DWgXAEDgKzWarn-HS_JU0TvYqs.</VersionId>
    <IsLatest>true</IsLatest>
    <LastModified>2009-12-10T16:38:11.000Z</LastModified>
    <Owner>
      <ID>75aa57f09aa0c8caeb4f8c2e99d10f8e7f8e9f76c078efc7c6caea54ba06a</ID>
    </Owner>
  </DeleteMarker>
  <Version>
    <Key>sourcekey</Key>
    <VersionId>wxxQ7ezLaL5N2Si1q66Sy6x0k7uHTUbp9qiiMxNg.</VersionId>
    <IsLatest>false</IsLatest>
    <LastModified>2009-12-10T16:37:44.000Z</LastModified>
    <ETag>"396fefe5f36d5ce46c7537efc978a360"</ETag>
    <Size>217</Size>
  </Version>
</ListVersionsResult>
Sample Request: Using key-marker and version-id-marker parameters

The following example returns objects starting at the specified key (key-marker) and version ID (version-id-marker).

GET /?versions&amp;key-marker=key3&amp;version-id-marker=t46ZenlYTZBnj HTTP/1.1
Host: bucket.s3.&lt;Region&gt;.amazonaws.com
Date: Wed, 28 Oct 2009 22:32:00 +0000
Authorization: signatureValue

Sample Response

This example illustrates one usage of ListObject Versions.

Sample Request: Using key-marker, version-id-marker, and max-keys

The following request returns up to three (the value of max-keys) objects starting with the key specified by key-marker and the version ID specified by version-id-marker.
Sample Response

This example illustrates one usage of ListObjectVersions.

```xml
<?xml version="1.0" encoding="UTF-8"?>
  <Name>mtp-versioning-fresh</Name>
  <Prefix/>
  <KeyMarker>key3</KeyMarker>
  <VersionIdMarker>null</VersionIdMarker>
  <NextKeyMarker>key3</NextKeyMarker>
  <NextVersionIdMarker>d-d309mf/jFrUmoQ0DBsVqmcMV15O1.</NextVersionIdMarker>
  <MaxKeys>3</MaxKeys>
  <IsTruncated>true</IsTruncated>
  <Version>
    <Key>key3</Key>
    <VersionId>8XECiENpj8pydEDJdd-_VRrvaGKAHoNgMNW7tg6UVI.</VersionId>
    <IsLatest>false</IsLatest>
    <LastModified>2009-12-09T00:18:23.000Z</LastModified>
    <ETag>"396fefef536d5ce46c7537ecf978a360"</ETag>
    <Size>217</Size>
    <Owner>
      <ID>75aa57f09a0c8caea54ba06a</ID>
    </Owner>
    <StorageClass>STANDARD</StorageClass>
  </Version>
  <Version>
    <Key>key3</Key>
    <VersionId>d-d309mf/jFrUmoQ0DBsVqmcMV15O1.</VersionId>
    <IsLatest>false</IsLatest>
    <LastModified>2009-12-09T00:18:08.000Z</LastModified>
    <ETag>"396fefef536d5ce46c7537ecf978a360"</ETag>
    <Size>217</Size>
    <Owner>
      <ID>75aa57f09a0c8caea54ba06a</ID>
    </Owner>
    <StorageClass>STANDARD</StorageClass>
  </Version>
</ListVersionsResult>
```

Sample Request: Using the delimiter and prefix parameters

Assume you have the following keys in your bucket, example-bucket.

- photos/2006/January/sample.jpg
- photos/2006/February/sample.jpg
- photos/2006/March/sample.jpg
- videos/2006/March/sample.wmv
- sample.jpg
The following GET versions request specifies the delimiter parameter with value "/".

```
GET /?versions&delimiter=/ HTTP/1.1
Host: example-bucket.s3.<Region>.amazonaws.com
Date: Wed, 02 Feb 2011 20:34:56 GMT
Authorization: authorization string
```

**Sample Response**

The list of keys from the specified bucket is shown in the following response.

The response returns the sample.jpg key in a `<Version>` element. However, because all the other keys contain the specified delimiter, a distinct substring, from the beginning of the key to the first occurrence of the delimiter, from each of these keys is returned in a `<CommonPrefixes>` element. The key substrings, photos/ and videos/, in the `<CommonPrefixes>` element indicate that there are one or more keys with these key prefixes.

This is a useful scenario if you use key prefixes for your objects to create a logical folder-like structure. In this case, you can interpret the result as the folders photos/ and videos/ have one or more objects.

```
  <Name>mvbucketwithversionon1</Name>
  <Prefix></Prefix>
  <KeyMarker></KeyMarker>
  <VersionIdMarker></VersionIdMarker>
  <MaxKeys>1000</MaxKeys>
  <Delimiter>/</Delimiter>
  <IsTruncated>false</IsTruncated>

  <Version>
    <Key>Sample.jpg</Key>
    <VersionId>toxMzQlBsGyGCz1YuMWmp90cdXLzqOCH</VersionId>
    <IsLatest>true</IsLatest>
    <LastModified>2011-02-02T18:46:20.000Z</LastModified>
    <ETag>"3305f2cfc46c0f04559748bb039d69ae"</ETag>
    <Size>3191</Size>
    <Owner>
      <ID>852b113e7a2f25102679df29b936c4393484be31bebcc</ID>
      <DisplayName>display-name</DisplayName>
    </Owner>
    <StorageClass>STANDARD</StorageClass>
  </Version>

  <CommonPrefixes>
    <Prefix>photos/</Prefix>
  </CommonPrefixes>
  <CommonPrefixes>
    <Prefix>videos/</Prefix>
  </CommonPrefixes>
</ListVersionsResult>
```

**Example**

In addition to the delimiter parameter, you can filter results by adding a prefix parameter as shown in the following request.
Example

In this case, the response will include only objects keys that start with the specified prefix. The value returned in the `<CommonPrefixes>` element is a substring from the beginning of the key to the first occurrence of the specified delimiter after the prefix.

```xml
<?xml version="1.0" encoding="UTF-8"?>
  <Name>example-bucket</Name>
  <Prefix>photos/2006/</Prefix>
  <KeyMarker></KeyMarker>
  <VersionIdMarker></VersionIdMarker>
  <MaxKeys>1000</MaxKeys>
  <Delimiter>/</Delimiter>
  <IsTruncated>false</IsTruncated>
  <Version>
    <Key>photos/2006/</Key>
    <VersionId>3U275dAA4gs82ZqOPHtJCUoI60krpCdY</VersionId>
    <IsLatest>true</IsLatest>
    <LastModified>2011-02-02T18:47:27.000Z</LastModified>
    <ETag>"d41d8cd98f00b204e9800998ecf8427e"</ETag>
    <Size>0</Size>
    <Owner>
      <ID>75aa57f09aa0c8caeab4f8c24e99d10f8e7faeebf76c078efc7c6caea54ba06a</ID>
      <DisplayName>display-name</DisplayName>
    </Owner>
    <StorageClass>STANDARD</StorageClass>
  </Version>
  <CommonPrefixes>
    <Prefix>photos/2006/February/</Prefix>
  </CommonPrefixes>
  <CommonPrefixes>
    <Prefix>photos/2006/January/</Prefix>
  </CommonPrefixes>
  <CommonPrefixes>
    <Prefix>photos/2006/March/</Prefix>
  </CommonPrefixes>
</ListVersionsResult>
```

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
• AWS SDK for Ruby V3
ListParts
Service: Amazon Simple Storage Service

Lists the parts that have been uploaded for a specific multipart upload. This operation must include the upload ID, which you obtain by sending the initiate multipart upload request (see CreateMultipartUpload). This request returns a maximum of 1,000 uploaded parts. The default number of parts returned is 1,000 parts. You can restrict the number of parts returned by specifying the max-parts request parameter. If your multipart upload consists of more than 1,000 parts, the response returns an IsTruncated field with the value of true, and a NextPartNumberMarker element. In subsequent ListParts requests you can include the part-number-marker query string parameter and set its value to the NextPartNumberMarker field value from the previous response.

For more information on multipart uploads, see Uploading Objects Using Multipart Upload.

For information on permissions required to use the multipart upload API, see Multipart Upload API and Permissions.

The following operations are related to ListParts:
• CreateMultipartUpload
• UploadPart
• CompleteMultipartUpload
• AbortMultipartUpload
• ListMultipartUploads

Request Syntax

GET /Key+?max-parts=MaxParts&part-number-marker=PartNumberMarker&uploadId=UploadId HTTP/1.1
Host: Bucket.s3.amazonaws.com
x-amz-request-payer: RequestPayer
x-amz-expected-bucket-owner: ExpectedBucketOwner

URI Request Parameters

The request uses the following URI parameters.

Bucket (p. 252)

The name of the bucket to which the parts are being uploaded.

When using this API with an access point, you must direct requests to the access point hostname. The access point hostname takes the form AccessPointName-AccountId.s3-accesspoint.Region.amazonaws.com. When using this operation with an access point through the AWS SDKs, you provide the access point ARN in place of the bucket name. For more information about access point ARNs, see Using Access Points in the Amazon Simple Storage Service Developer Guide.

When using this API with Amazon S3 on Outposts, you must direct requests to the S3 on Outposts hostname. The S3 on Outposts hostname takes the form AccessPointName-AccountId.outpostID.s3-outposts.Region.amazonaws.com. When using this operation using S3 on Outposts through the AWS SDKs, you provide the Outposts bucket ARN in place of the bucket name. For more information about S3 on Outposts ARNs, see Using S3 on Outposts in the Amazon Simple Storage Service Developer Guide.

Required: Yes
Key (p. 252)

Object key for which the multipart upload was initiated.
Length Constraints: Minimum length of 1.
Required: Yes

max-parts (p. 252)

Sets the maximum number of parts to return.

part-number-marker (p. 252)

Specifies the part after which listing should begin. Only parts with higher part numbers will be listed.

uploadId (p. 252)

Upload ID identifying the multipart upload whose parts are being listed.
Required: Yes

x-amz-expected-bucket-owner (p. 252)

The account id of the expected bucket owner. If the bucket is owned by a different account, the request will fail with an HTTP 403 (Access Denied) error.

x-amz-request-payer (p. 252)

Confirms that the requester knows that they will be charged for the request. Bucket owners need not specify this parameter in their requests. For information about downloading objects from requester pays buckets, see Downloading Objects in Requestor Pays Buckets in the Amazon S3 Developer Guide.

Valid Values: requester

Request Body

The request does not have a request body.

Response Syntax

```
HTTP/1.1 200
x-amz-abort-date: AbortDate
x-amz-abort-rule-id: AbortRuleId
x-amz-request-charged: RequestCharged
<?xml version="1.0" encoding="UTF-8"?>
<ListPartsResult>
  <Bucket>string</Bucket>
  <Key>string</Key>
  <UploadId>string</UploadId>
  <PartNumberMarker>integer</PartNumberMarker>
  <NextPartNumberMarker>integer</NextPartNumberMarker>
  <MaxParts>integer</MaxParts>
  <IsTruncated>boolean</IsTruncated>
  <Part>
    <ETag>string</ETag>
    <LastModified>timestamp</LastModified>
    <PartNumber>integer</PartNumber>
    <Size>integer</Size>
  </Part>
  ...
  <Initiator>
    <DisplayName>string</DisplayName>
  </Initiator>
</ListPartsResult>
```
Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The response returns the following HTTP headers.

**x-amz-abort-date (p. 253)**

If the bucket has a lifecycle rule configured with an action to abort incomplete multipart uploads and the prefix in the lifecycle rule matches the object name in the request, then the response includes this header indicating when the initiated multipart upload will become eligible for abort operation. For more information, see Aborting Incomplete Multipart Uploads Using a Bucket Lifecycle Policy.

The response will also include the x-amz-abort-rule-id header that will provide the ID of the lifecycle configuration rule that defines this action.

**x-amz-abort-rule-id (p. 253)**

This header is returned along with the x-amz-abort-date header. It identifies applicable lifecycle configuration rule that defines the action to abort incomplete multipart uploads.

**x-amz-request-charged (p. 253)**

If present, indicates that the requester was successfully charged for the request.

Valid Values: requester

The following data is returned in XML format by the service.

**ListPartsResult (p. 253)**

Root level tag for the ListPartsResult parameters.

Required: Yes

**Bucket (p. 253)**

The name of the bucket to which the multipart upload was initiated.

Type: String

**Initiator (p. 253)**

Container element that identifies who initiated the multipart upload. If the initiator is an AWS account, this element provides the same information as the Owner element. If the initiator is an IAM User, this element provides the user ARN and display name.

Type: initiator (p. 570) data type

**IsTruncated (p. 253)**

Indicates whether the returned list of parts is truncated. A true value indicates that the list was truncated. A list can be truncated if the number of parts exceeds the limit returned in the MaxParts element.
Type: Boolean

**Key (p. 253)**

Object key for which the multipart upload was initiated.

Type: String

Length Constraints: Minimum length of 1.

**MaxParts (p. 253)**

Maximum number of parts that were allowed in the response.

Type: Integer

**NextPartNumberMarker (p. 253)**

When a list is truncated, this element specifies the last part in the list, as well as the value to use for the part-number-marker request parameter in a subsequent request.

Type: Integer

**Owner (p. 253)**

Container element that identifies the object owner, after the object is created. If multipart upload is initiated by an IAM user, this element provides the parent account ID and display name.

Type: Owner (p. 617) data type

**Part (p. 253)**

Container for elements related to a particular part. A response can contain zero or more Part elements.

Type: Array of Part (p. 621) data types

**PartNumberMarker (p. 253)**

When a list is truncated, this element specifies the last part in the list, as well as the value to use for the part-number-marker request parameter in a subsequent request.

Type: Integer

**StorageClass (p. 253)**

Class of storage (STANDARD or REDUCED_REDUNDANCY) used to store the uploaded object.

Type: String

Valid Values:
- STANDARD
- REDUCED_REDUNDANCY
- STANDARD_IA
- ONEZONE_IA
- INTELLIGENT_TIERING
- GLACIER
- DEEP_ARCHIVE
- OUTPOSTS

**UploadId (p. 253)**

Upload ID identifying the multipart upload whose parts are being listed.

Type: String

---

**Examples**

**Sample Request**

Assume you have uploaded parts with sequential part numbers starting with 1. The following List Parts request specifies max-parts and part-number-marker query parameters. The request lists the first two parts that follow part number 1, that is, you will get parts 2 and 3 in the response. If more parts exist, the result is a truncated result and therefore the response will return an IsTruncated element.
with the value true. The response will also return the NextPartNumberMarker element with the value 3, which should be used for the value of the part-number-marker request query string parameter in the next ListParts request.

```
GET /example-object?
uploadId=XXBsbeb2FkIrIcIGzvciBibHZpZmhcncyVcdSltb3Z5tMnRszEEwbG9hZA&max-parts=2&part-number-marker=1 HTTP/1.1
Host: example-bucket.s3.<Region>.amazonaws.com
Date: Mon, 1 Nov 2010 20:34:56 GMT
Authorization: authorization string

Sample Response

This example illustrates one usage of ListParts.

```
<?xml version="1.0" encoding="UTF-8"?>
<ListPartsResult xmlns="http://s3.amazonaws.com/doc/2006-03-01/">
  <Bucket>example-bucket</Bucket>
  <Key>example-object</Key>
  <UploadId>XXBsb2FkIElEIGZvciBlbHZpZmhcncyVcdSltb3Z5tMnRszEEwbG9hZA</UploadId>
  <Initiator>
    <ID>arn:aws:iam::111122223333:user/some-user-11116a31-17b5-4fb7-9df5-b288870f11xx</ID>
    <DisplayName>umat-user-11116a31-17b5-4fb7-9df5-b288870f11xx</DisplayName>
  </Initiator>
  <Owner>
    <ID>75aa57f09aa0c8caab482c4e99d10f8e7faeeb776c078ef8c76a06a</ID>
    <DisplayName>someName</DisplayName>
  </Owner>
  <StorageClass>STANDARD</StorageClass>
  <PartNumberMarker>1</PartNumberMarker>
  <NextPartNumberMarker>3</NextPartNumberMarker>
  <MaxParts>2</MaxParts>
  <IsTruncated>true</IsTruncated>
  <Part>
    <PartNumber>2</PartNumber>
    <LastModified>2010-11-10T20:48:34.000Z</LastModified>
    <ETag>"7778af83f66abc1fae8477f296d394"</ETag>
    <Size>10485760</Size>
  </Part>
  <Part>
    <PartNumber>3</PartNumber>
    <LastModified>2010-11-10T20:48:33.000Z</LastModified>
    <ETag>"aaaa18db4cc2f5cedef654fccc4a4x8"</ETag>
    <Size>10485760</Size>
  </Part>
</ListPartsResult>
```

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:
• AWS Command Line Interface
• AWS SDK for .NET
• AWS SDK for C++
• AWS SDK for Go
• AWS SDK for Java
• AWS SDK for JavaScript
• AWS SDK for PHP V3
• AWS SDK for Python
• AWS SDK for Ruby V3
PutBucketAccelerateConfiguration

Service: Amazon Simple Storage Service

Sets the accelerate configuration of an existing bucket. Amazon S3 Transfer Acceleration is a bucket-level feature that enables you to perform faster data transfers to Amazon S3.

To use this operation, you must have permission to perform the s3:PutAccelerateConfiguration action. The bucket owner has this permission by default. The bucket owner can grant this permission to others. For more information about permissions, see Permissions Related to Bucket Subresource Operations and Managing Access Permissions to Your Amazon S3 Resources.

The Transfer Acceleration state of a bucket can be set to one of the following two values:

- Enabled – Enables accelerated data transfers to the bucket.
- Suspended – Disables accelerated data transfers to the bucket.

The GetBucketAccelerateConfiguration operation returns the transfer acceleration state of a bucket.

After setting the Transfer Acceleration state of a bucket to Enabled, it might take up to thirty minutes before the data transfer rates to the bucket increase.

The name of the bucket used for Transfer Acceleration must be DNS-compliant and must not contain periods ("."). For more information about transfer acceleration, see Transfer Acceleration.

The following operations are related to PutBucketAccelerateConfiguration:

- GetBucketAccelerateConfiguration
- CreateBucket

Request Syntax

```
PUT /?accelerate HTTP/1.1
Host: Bucket.s3.amazonaws.com
x-amz-expected-bucket-owner: ExpectedBucketOwner
<?xml version="1.0" encoding="UTF-8"?>
  <Status>string</Status>
</AccelerateConfiguration>
```

URI Request Parameters

The request uses the following URI parameters.

**Bucket (p. 258)**

The name of the bucket for which the accelerate configuration is set.

Required: Yes

**x-amz-expected-bucket-owner (p. 258)**

The account id of the expected bucket owner. If the bucket is owned by a different account, the request will fail with an HTTP 403 (Access Denied) error.

Request Body

The request accepts the following data in XML format.
**AccelerateConfiguration (p. 258)**

Root level tag for the AccelerateConfiguration parameters.

Required: Yes

**Status (p. 258)**

Specifies the transfer acceleration status of the bucket.

Type: String

Valid Values: Enabled | Suspended

Required: No

**Response Syntax**

```
HTTP/1.1 200
```

**Response Elements**

If the action is successful, the service sends back an HTTP 200 response with an empty HTTP body.

**Examples**

**Sample Request: Add transfer acceleration configuration to set acceleration status**

The following is an example of a PUT /?accelerate request that enables transfer acceleration for the bucket named examplebucket.

```
PUT /?accelerate HTTP/1.1
Host: examplebucket.s3.<Region>.amazonaws.com
Date: Mon, 11 Apr 2016 12:00:00 GMT
Authorization: authorization string
Content-Type: text/plain
Content-Length: length

  <Status>Enabled</Status>
</AccelerateConfiguration>
```

**Sample Response**

This example illustrates one usage of PutBucketAccelerateConfiguration.

```
HTTP/1.1 200 OK
x-amz-id-2: YgIPIfBiKa2bj0KMg95r/02o3em2U4dzzD4rcKCHQUAdQkf3ShJT0OpXUueF6Qko
x-amz-request-id: 236A8905248E5A01
Date: Mon, 11 Apr 2016 12:00:00 GMT
Content-Length: 0
Server: AmazonS3
```

**See Also**

For more information about using this API in one of the language-specific AWS SDKs, see the following:
AWS Command Line Interface
AWS SDK for .NET
AWS SDK for C++
AWS SDK for Go
AWS SDK for Java
AWS SDK for JavaScript
AWS SDK for PHP V3
AWS SDK for Python
AWS SDK for Ruby V3
PutBucketAcl
Service: Amazon Simple Storage Service

Sets the permissions on an existing bucket using access control lists (ACL). For more information, see Using ACLs. To set the ACL of a bucket, you must have WRITE_ACP permission.

You can use one of the following two ways to set a bucket's permissions:

- Specify the ACL in the request body
- Specify permissions using request headers

**Note**
You cannot specify access permission using both the body and the request headers.

Depending on your application needs, you may choose to set the ACL on a bucket using either the request body or the headers. For example, if you have an existing application that updates a bucket ACL using the request body, then you can continue to use that approach.

**Access Permissions**

You can set access permissions using one of the following methods:

- Specify a canned ACL with the x-amz-acl request header. Amazon S3 supports a set of predefined ACLs, known as **canned ACLs**. Each canned ACL has a predefined set of grantees and permissions. Specify the canned ACL name as the value of x-amz-acl. If you use this header, you cannot use other access control-specific headers in your request. For more information, see Canned ACL.

- Specify access permissions explicitly with the x-amz-grant-read, x-amz-grant-read-acp, x-amz-grant-write-acp, and x-amz-grant-full-control headers. When using these headers, you specify explicit access permissions and grantees (AWS accounts or Amazon S3 groups) who will receive the permission. If you use these ACL-specific headers, you cannot use the x-amz-acl header to set a canned ACL. These parameters map to the set of permissions that Amazon S3 supports in an ACL. For more information, see Access Control List (ACL) Overview.

You specify each grantee as a type=value pair, where the type is one of the following:

- **id** – if the value specified is the canonical user ID of an AWS account
- **uri** – if you are granting permissions to a predefined group
- **emailAddress** – if the value specified is the email address of an AWS account

**Note**
Using email addresses to specify a grantee is only supported in the following AWS Regions:

- US East (N. Virginia)
- US West (N. California)
- US West (Oregon)
- Asia Pacific (Singapore)
- Asia Pacific (Sydney)
- Asia Pacific (Tokyo)
- Europe (Ireland)
- South America (São Paulo)

For a list of all the Amazon S3 supported Regions and endpoints, see Regions and Endpoints in the AWS General Reference.

For example, the following x-amz-grant-write header grants create, overwrite, and delete objects permission to LogDelivery group predefined by Amazon S3 and two AWS accounts identified by their email addresses.
You can use either a canned ACL or specify access permissions explicitly. You cannot do both.

Grantee Values

You can specify the person (grantee) to whom you're assigning access rights (using request elements) in the following ways:

- By the person's ID:

  <Grantee xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:type="CanonicalUser"><ID><ID></ID><DisplayName><GranteesEmail/></DisplayName></Grantee>

  DisplayName is optional and ignored in the request

- By URI:


- By Email address:

  <Grantee xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:type="AmazonCustomerByEmail"><EmailAddress><Grantees@email.com/></EmailAddress></Grantee>

  The grantee is resolved to the CanonicalUser and, in a response to a GET Object acl request, appears as the CanonicalUser.

  Note
  Using email addresses to specify a grantee is only supported in the following AWS Regions:
  - US East (N. Virginia)
  - US West (N. California)
  - US West (Oregon)
  - Asia Pacific (Singapore)
  - Asia Pacific (Sydney)
  - Asia Pacific (Tokyo)
  - Europe (Ireland)
  - South America (São Paulo)
  For a list of all the Amazon S3 supported Regions and endpoints, see Regions and Endpoints in the AWS General Reference.

Related Resources

- CreateBucket
- DeleteBucket
- GetObjectAcl

Request Syntax

PUT /?acl HTTP/1.1
URI Request Control Parameters

The request uses the following URI parameters.

**Bucket (p. 262)**

The bucket to which to apply the ACL.

Required: Yes

**Content-MD5 (p. 262)**

This header has been deprecated.

The base64-encoded 128-bit MD5 digest of the data. This header must be used as a message integrity check to verify that the request body was not corrupted in transit. For more information, go to RFC 1864.

**x-amz-acl (p. 262)**

The canned ACL to apply to the bucket.

Valid Values: private | public-read | public-read-write | authenticated-read

**x-amz-expected-bucket-owner (p. 262)**

The account id of the expected bucket owner. If the bucket is owned by a different account, the request will fail with an HTTP 403 (Access Denied) error.

**x-amz-grant-full-control (p. 262)**

Allows grantee the read, write, read ACP, and write ACP permissions on the bucket.

**x-amz-grant-read (p. 262)**

Allows grantee to list the objects in the bucket.
x-amz-grant-read-acp (p. 262)

Allows grantee to read the bucket ACL.

x-amz-grant-write (p. 262)

Allows grantee to create, overwrite, and delete any object in the bucket.

x-amz-grant-write-acp (p. 262)

Allows grantee to write the ACL for the applicable bucket.

Request Body

The request accepts the following data in XML format.

AccessControlPolicy (p. 262)

Root level tag for the AccessControlPolicy parameters.

Required: Yes

Grants (p. 262)

A list of grants.

Type: Array of Grant (p. 566) data types

Required: No

Owner (p. 262)

Container for the bucket owner's display name and ID.

Type: Owner (p. 617) data type

Required: No

Response Syntax

```
HTTP/1.1 200
```

Response Elements

If the action is successful, the service sends back an HTTP 200 response with an empty HTTP body.

Examples

Sample Request: Access permissions specified in the body

The following request grants access permission to the existing examplebucket bucket. The request specifies the ACL in the body. In addition to granting full control to the bucket owner, the XML specifies the following grants.

- Grant the AllUsers group READ permission on the bucket.
- Grant the LogDelivery group WRITE permission on the bucket.
- Grant an AWS account, identified by email address, WRITE_ACP permission.
- Grant an AWS account, identified by canonical user ID, READ_ACP permission.
Sample Response

This example illustrates one usage of PutBucketAcl.

HTTP/1.1 200 OK
x-amz-id-2: NxoQ3PNI1MHXXGwjdjg15LgUoAmPVmG0xtZw2sxeFgLhp1vcyouXDrqcUaCWXc0K
x-amz-request-id: C651BC9B4E1BD401
Date: Thu, 12 Apr 2012 20:04:28 GMT
Content-Length: 0
Server: AmazonS3
Sample Request: Access permissions specified using headers

The following request uses ACL-specific request headers to grant the following permissions:

- Write permission to the Amazon S3 LogDelivery group and an AWS account identified by the email xyz@amazon.com.
- Read permission to the Amazon S3 AllUsers group

```
PUT ?acl HTTP/1.1
Host: examplebucket.s3.<Region>.amazonaws.com
x-amz-date: Sun, 29 Apr 2012 22:00:57 GMT
x-amz-grant-write: uri="http://acs.amazonaws.com/groups/s3/LogDelivery",
  emailAddress="xyz@amazon.com"
x-amz-grant-read: uri="http://acs.amazonaws.com/groups/global/AllUsers"
Accept: */*
Authorization: authorization string
```

Sample Response

This example illustrates one usage of PutBucketAcl.

```
HTTP/1.1 200 OK
x-amz-id-2: Ow9iImt23VF9s6QofOTDze1F7mrryz7d04Mw23FQcI40205Zw28Zn+d340/RytoQ
x-amz-request-id: A6A8F01A38EC7138
Date: Sun, 29 Apr 2012 22:01:10 GMT
Content-Length: 0
Server: AmazonS3
```

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3
PutBucketAnalyticsConfiguration
Service: Amazon Simple Storage Service

Sets an analytics configuration for the bucket (specified by the analytics configuration ID). You can have up to 1,000 analytics configurations per bucket.

You can choose to have storage class analysis export analysis reports sent to a comma-separated values (CSV) flat file. See the DataExport request element. Reports are updated daily and are based on the object filters that you configure. When selecting data export, you specify a destination bucket and an optional destination prefix where the file is written. You can export the data to a destination bucket in a different account. However, the destination bucket must be in the same Region as the bucket that you are making the PUT analytics configuration to. For more information, see Amazon S3 Analytics – Storage Class Analysis.

Important
You must create a bucket policy on the destination bucket where the exported file is written to grant permissions to Amazon S3 to write objects to the bucket. For an example policy, see Granting Permissions for Amazon S3 Inventory and Storage Class Analysis.

To use this operation, you must have permissions to perform the s3:PutAnalyticsConfiguration action. The bucket owner has this permission by default. The bucket owner can grant this permission to others. For more information about permissions, see Permissions Related to Bucket Subresource Operations and Managing Access Permissions to Your Amazon S3 Resources.

Special Errors

- HTTP Error: HTTP 400 Bad Request
  - Code: InvalidArgument
  - Cause: Invalid argument.
- HTTP Error: HTTP 400 Bad Request
  - Code: TooManyConfigurations
  - Cause: You are attempting to create a new configuration but have already reached the 1,000-configuration limit.
- HTTP Error: HTTP 403 Forbidden
  - Code: AccessDenied
  - Cause: You are not the owner of the specified bucket, or you do not have the s3:PutAnalyticsConfiguration bucket permission to set the configuration on the bucket.

Related Resources

- GetBucketAnalyticsConfiguration
- DeleteBucketAnalyticsConfiguration
- ListBucketAnalyticsConfigurations

Request Syntax

```xml
PUT /?analytics&id=Id HTTP/1.1
Host: Bucket.s3.amazonaws.com
x-amz-expected-bucket-owner: ExpectedBucketOwner
<?xml version="1.0" encoding="UTF-8"?>
<AnalyticsConfiguration xmlns="http://s3.amazonaws.com/doc/2006-03-01/">
  <Id>string</Id>
  <Filter>
    <And>
      <Prefix>string</Prefix>
    </And>
  </Filter>
</AnalyticsConfiguration>
```
<Tag>
  <Key>string</Key>
  <Value>string</Value>
</Tag>
...
</And>
<Prefix>string</Prefix>
<Tag>
  <Key>string</Key>
  <Value>string</Value>
</Tag>
</Filter>
<StorageClassAnalysis>
  <DataExport>
    <Destination>
      <S3BucketDestination>
        <Bucket>string</Bucket>
        <BucketAccountId>string</BucketAccountId>
        <Prefix>string</Prefix>
      </S3BucketDestination>
    </Destination>
    <OutputSchemaVersion>string</OutputSchemaVersion>
  </DataExport>
</StorageClassAnalysis>
</AnalyticsConfiguration>

**URI Request Parameters**

The request uses the following URI parameters.

**Bucket (p. 267)**

The name of the bucket to which an analytics configuration is stored.

Required: Yes

**id (p. 267)**

The ID that identifies the analytics configuration.

Required: Yes

**x-amz-expected-bucket-owner (p. 267)**

The account id of the expected bucket owner. If the bucket is owned by a different account, the request will fail with an HTTP 403 (Access Denied) error.

**Request Body**

The request accepts the following data in XML format.

**AnalyticsConfiguration (p. 267)**

Root level tag for the AnalyticsConfiguration parameters.

Required: Yes

**Filter (p. 267)**

The filter used to describe a set of objects for analyses. A filter must have exactly one prefix, one tag, or one conjunction (AnalyticsAndOperator). If no filter is provided, all objects will be considered in any analysis.
Type: AnalyticsFilter (p. 521) data type

Required: No

Id (p. 267)

The ID that identifies the analytics configuration.

Type: String

Required: Yes

StorageClassAnalysis (p. 267)

Contains data related to access patterns to be collected and made available to analyze the tradeoffs between different storage classes.

Type: StorageClassAnalysis (p. 664) data type

Required: Yes

Response Syntax

HTTP/1.1 200

Response Elements

If the action is successful, the service sends back an HTTP 200 response with an empty HTTP body.

Examples

Example 1: Creating an analytics configuration

The following PUT request for the bucket examplebucket creates a new or replaces an existing analytics configuration with the ID report1. The configuration is defined in the request body.

PUT /?analytics&id=report1 HTTP/1.1
Host: examplebucket.s3.<Region>.amazonaws.com
Date: Mon, 31 Oct 2016 12:00:00 GMT
Authorization: authorization string
Content-Length: length

<?xml version="1.0" encoding="UTF-8"?>
<AnalyticsConfiguration xmlns="http://s3.amazonaws.com/doc/2006-03-01/"
  <Id>report1</Id>
  <Filter>
    <And>
      <Prefix>images/</Prefix>
      <Tag>
        <Key>dog</Key>
        <Value>corgi</Value>
      </Tag>
    </And>
  </Filter>
  <StorageClassAnalysis>
    <DataExport>
      <OutputSchemaVersion>V_1</OutputSchemaVersion>
      <Destination>
        <S3BucketDestination>
          <Format>CSV</Format>
        </S3BucketDestination>
      </Destination>
    </DataExport>
  </StorageClassAnalysis>
</AnalyticsConfiguration>
Sample Response

This example illustrates one usage of PutBucketAnalyticsConfiguration.

HTTP/1.1 200 OK
x-amz-id-2: YgIPIfBiKa2bj0KMg95r/0zo3emzU4dzsD4rcKCHQUAdQkf3ShJTOOpXUueF6QKo
x-amz-request-id: 236A8905248E5A01
Date: Mon, 31 Oct 2016 12:00:00 GMT
Content-Length: 0
Server: AmazonS3

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3
PutBucketCors
Service: Amazon Simple Storage Service

Sets the cors configuration for your bucket. If the configuration exists, Amazon S3 replaces it.

To use this operation, you must be allowed to perform the s3:PutBucketCORS action. By default, the bucket owner has this permission and can grant it to others.

You set this configuration on a bucket so that the bucket can service cross-origin requests. For example, you might want to enable a request whose origin is http://www.example.com to access your Amazon S3 bucket at my.example.bucket.com by using the browser's XMLHttpRequest capability.

To enable cross-origin resource sharing (CORS) on a bucket, you add the cors subresource to the bucket. The cors subresource is an XML document in which you configure rules that identify origins and the HTTP methods that can be executed on your bucket. The document is limited to 64 KB in size.

When Amazon S3 receives a cross-origin request (or a pre-flight OPTIONS request) against a bucket, it evaluates the cors configuration on the bucket and uses the first CORSRule rule that matches the incoming browser request to enable a cross-origin request. For a rule to match, the following conditions must be met:

- The request's Origin header must match AllowedOrigin elements.
- The request method (for example, GET, PUT, HEAD, and so on) or the Access-Control-Request-Method header in case of a pre-flight OPTIONS request must be one of the AllowedMethod elements.
- Every header specified in the Access-Control-Request-Headers request header of a pre-flight request must match an AllowedHeader element.

For more information about CORS, go to Enabling Cross-Origin Resource Sharing in the Amazon Simple Storage Service Developer Guide.

Related Resources
- GetBucketCors
- DeleteBucketCors
- RESTOPTIONSObject

Request Syntax

PUT /?cors HTTP/1.1
Host: Bucket.s3.amazonaws.com
Content-MD5: ContentMD5
x-amz-expected-bucket-owner: ExpectedBucketOwner
<?xml version="1.0" encoding="UTF-8"?>
<CORSConfiguration xmlns="http://s3.amazonaws.com/doc/2006-03-01/">
    <CORSRule>
        <AllowedHeader>string</AllowedHeader>
        ...
        <AllowedMethod>string</AllowedMethod>
        ...
        <AllowedOrigin>string</AllowedOrigin>
        ...
        <ExposeHeader>string</ExposeHeader>
        ...
        <MaxAgeSeconds>integer</MaxAgeSeconds>
    </CORSRule>
    ...
</CORSConfiguration>
URI Request Parameters

The request uses the following URI parameters.

**Bucket (p. 271)**

Specifies the bucket impacted by the corsconfiguration.

Required: Yes

**Content-MD5 (p. 271)**

*This header has been deprecated.*

The base64-encoded 128-bit MD5 digest of the data. This header must be used as a message integrity check to verify that the request body was not corrupted in transit. For more information, go to RFC 1864.

**x-amz-expected-bucket-owner (p. 271)**

The account id of the expected bucket owner. If the bucket is owned by a different account, the request will fail with an HTTP 403 (Access Denied) error.

Request Body

The request accepts the following data in XML format.

**CORSConfiguration (p. 271)**

Root level tag for the CORSConfiguration parameters.

Required: Yes

**CORSRule (p. 271)**

A set of origins and methods (cross-origin access that you want to allow). You can add up to 100 rules to the configuration.

Type: Array of CORSRule (p. 536) data types

Required: Yes

Response Syntax

<table>
<thead>
<tr>
<th>HTTP/1.1 200</th>
</tr>
</thead>
</table>

Response Elements

If the action is successful, the service sends back an HTTP 200 response with an empty HTTP body.

Examples

**Example: Cors configuration on a bucket with two rules**

- The first CORSRule allows cross-origin PUT, POST, and DELETE requests whose origin is http://www.example.com origins. The rule also allows all headers in a pre-flight OPTIONS request through the Access-Control-Request-Headers header. Therefore, in response to any pre-flight OPTIONS request, Amazon S3 will return any requested headers.
The second rule allows cross-origin GET requests from all the origins. The '*' wildcard character refers to all origins.

```
<CORSConfiguration>
  <CORSRule>
    <AllowedOrigin>http://www.example.com</AllowedOrigin>
    <AllowedMethod>PUT</AllowedMethod>
    <AllowedMethod>POST</AllowedMethod>
    <AllowedMethod>DELETE</AllowedMethod>
    <AllowedHeader>*</AllowedHeader>
  </CORSRule>
  <CORSRule>
    <AllowedOrigin>*</AllowedOrigin>
    <AllowedMethod>GET</AllowedMethod>
  </CORSRule>
</CORSConfiguration>
```

**Example:** Cors configuration allows cross-origin PUT and POST requests from http://www.example.com

The cors configuration also allows additional optional configuration parameters as shown in the following cors configuration on a bucket. For example,

In the preceding configuration, `CORSRule` includes the following additional optional parameters:

- **MaxAgeSeconds**—Specifies the time in seconds that the browser will cache an Amazon S3 response to a pre-flight OPTIONS request for the specified resource. In this example, this parameter is 3000 seconds. Caching enables the browsers to avoid sending pre-flight OPTIONS request to Amazon S3 for repeated requests.
- **ExposeHeader**—Identifies the response header (in this case `x-amz-server-side-encryption`) that you want customers to be able to access from their applications (for example, from a JavaScript XMLHttpRequest object).

```
<CORSConfiguration>
  <CORSRule>
    <AllowedOrigin>http://www.example.com</AllowedOrigin>
    <AllowedMethod>PUT</AllowedMethod>
    <AllowedMethod>POST</AllowedMethod>
    <AllowedMethod>DELETE</AllowedMethod>
    <MaxAgeSeconds>3000</MaxAgeSeconds>
    <ExposeHeader>x-amz-server-side-encryption</ExposeHeader>
  </CORSRule>
</CORSConfiguration>
```

**See Also**

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3
PutBucketEncryption

Service: Amazon Simple Storage Service

This implementation of the **PUT** operation uses the encryption subresource to set the default encryption state of an existing bucket.

This implementation of the **PUT** operation sets default encryption for a bucket using server-side encryption with Amazon S3-managed keys (SSE-S3) or customer master keys stored in AWS KMS (SSE-KMS). For information about the Amazon S3 default encryption feature, see Amazon S3 Default Bucket Encryption.

**Important**

This operation requires AWS Signature Version 4. For more information, see Authenticating Requests (AWS Signature Version 4).

To use this operation, you must have permissions to perform the **s3:PutEncryptionConfiguration** action. The bucket owner has this permission by default. The bucket owner can grant this permission to others. For more information about permissions, see Permissions Related to Bucket Subresource Operations and Managing Access Permissions to Your Amazon S3 Resources in the Amazon Simple Storage Service Developer Guide.

Related Resources

- GetBucketEncryption
- DeleteBucketEncryption

Request Syntax

```
PUT /?encryption HTTP/1.1
Host: Bucket.s3.amazonaws.com
Content-MD5: ContentMD5
x-amz-expected-bucket-owner: ExpectedBucketOwner
  <Rule>
    <ApplyServerSideEncryptionByDefault>
      <KMSMasterKeyID>string</KMSMasterKeyID>
      <SSEAlgorithm>string</SSEAlgorithm>
    </ApplyServerSideEncryptionByDefault>
  </Rule>
</ServerSideEncryptionConfiguration>
```

URI Request Parameters

The request uses the following URI parameters.

**Bucket** *(p. 275)*

Specifies default encryption for a bucket using server-side encryption with Amazon S3-managed keys (SSE-S3) or customer master keys stored in AWS KMS (SSE-KMS). For information about the Amazon S3 default encryption feature, see Amazon S3 Default Bucket Encryption in the Amazon Simple Storage Service Developer Guide.

Required: Yes

**Content-MD5** *(p. 275)*

*This header has been deprecated.*
The base64-encoded 128-bit MD5 digest of the server-side encryption configuration. This parameter is auto-populated when using the command from the CLI.

**x-amz-expected-bucket-owner (p. 275)**

The account id of the expected bucket owner. If the bucket is owned by a different account, the request will fail with an HTTP 403 (Access Denied) error.

**Request Body**

The request accepts the following data in XML format.

**ServerSideEncryptionConfiguration (p. 275)**

Root level tag for the ServerSideEncryptionConfiguration parameters.

Required: Yes

**Rule (p. 275)**

Container for information about a particular server-side encryption configuration rule.

Type: Array of ServerSideEncryptionRule (p. 657) data types

Required: Yes

**Response Syntax**

```
HTTP/1.1 200
```

**Response Elements**

If the action is successful, the service sends back an HTTP 200 response with an empty HTTP body.

**Examples**

In the request, you specify the encryption configuration in the request body. The encryption configuration is specified as XML, as shown in the following examples that show setting encryption using SSE-S3 or SSE-KMS.

**Request Body for Setting SSE-S3**

This example illustrates one usage of PutBucketEncryption.

```
<ApplyServerSideEncryptionByDefault>
  <SSEAlgorithm>AES256</SSEAlgorithm>
</ApplyServerSideEncryptionByDefault>
</Rule>
</ServerSideEncryptionConfiguration>
```

**Request Body for Setting SSE-KMS**

This example illustrates one usage of PutBucketEncryption.
Set the Default Encryption Configuration for an S3 Bucket

The following is an example of a PUT /? encryption request that specifies to use AWS KMS encryption.

```
PUT /?encryption HTTP/1.1
Host: examplebucket.<Region>s3.amazonaws.com
Date: Wed, 06 Sep 2017 12:00:00 GMT
Authorization: authorization
Content-Length: length

  <Rule>
    <ApplyServerSideEncryptionByDefault>
      <SSEAlgorithm>aws:kms</SSEAlgorithm>
      <KMSMasterKeyID>arn:aws:kms:us-east-1:1234/5678example</KMSMasterKeyID>
    </ApplyServerSideEncryptionByDefault>
  </Rule>
</ServerSideEncryptionConfiguration>
```

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3
PutBucketIntelligentTieringConfiguration

Service: Amazon Simple Storage Service

Puts a S3 Intelligent-Tiering configuration to the specified bucket.

The S3 Intelligent-Tiering storage class is designed to optimize storage costs by automatically moving data to the most cost-effective storage access tier, without additional operational overhead. S3 Intelligent-Tiering delivers automatic cost savings by moving data between access tiers, when access patterns change.

The S3 Intelligent-Tiering storage class is suitable for objects larger than 128 KB that you plan to store for at least 30 days. If the size of an object is less than 128 KB, it is not eligible for auto-tiering. Smaller objects can be stored, but they are always charged at the frequent access tier rates in the S3 Intelligent-Tiering storage class.

If you delete an object before the end of the 30-day minimum storage duration period, you are charged for 30 days. For more information, see Storage class for automatically optimizing frequently and infrequently accessed objects.

Operations related to PutBucketIntelligentTieringConfiguration include:

- DeleteBucketIntelligentTieringConfiguration
- GetBucketIntelligentTieringConfiguration
- ListBucketIntelligentTieringConfigurations

Request Syntax

```
PUT /?intelligent-tiering&id=Id HTTP/1.1
Host: Bucket.s3.amazonaws.com
<?xml version="1.0" encoding="UTF-8"?>
  <Id>string</Id>
  <Filter>
    <And>
      <Prefix>string</Prefix>
      <Tag>
        <Key>string</Key>
        <Value>string</Value>
      </Tag>
      ...
    </And>
    <Prefix>string</Prefix>
    <Tag>
      <Key>string</Key>
      <Value>string</Value>
    </Tag>
  </Filter>
  <Status>string</Status>
  <Tiering>
    <AccessTier>string</AccessTier>
    <Days>integer</Days>
  </Tiering>
  ...
</IntelligentTieringConfiguration>
```

URI Request Parameters

The request uses the following URI parameters.
Bucket (p. 278)
The name of the Amazon S3 bucket whose configuration you want to modify or retrieve.
Required: Yes

id (p. 278)
The ID used to identify the S3 Intelligent-Tiering configuration.
Required: Yes

Request Body
The request accepts the following data in XML format.

IntelligentTieringConfiguration (p. 278)
Root level tag for the IntelligentTieringConfiguration parameters.
Required: Yes

Filter (p. 278)
Specifies a bucket filter. The configuration only includes objects that meet the filter's criteria.
Type: IntelligentTieringFilter (p. 574) data type
Required: No

Id (p. 278)
The ID used to identify the S3 Intelligent-Tiering configuration.
Type: String
Required: Yes

Status (p. 278)
Specifies the status of the configuration.
Type: String
Valid Values: Enabled | Disabled
Required: Yes

Tiering (p. 278)
Specifies the S3 Intelligent-Tiering storage class tier of the configuration.
Type: Array of Tiering (p. 669) data types
Required: Yes

Response Syntax
HTTP/1.1 200

Response Elements
If the action is successful, the service sends back an HTTP 200 response with an empty HTTP body.
See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3
PutBucketInventoryConfiguration
Service: Amazon Simple Storage Service

This implementation of the PUT operation adds an inventory configuration (identified by the inventory ID) to the bucket. You can have up to 1,000 inventory configurations per bucket.

Amazon S3 inventory generates inventories of the objects in the bucket on a daily or weekly basis, and the results are published to a flat file. The bucket that is inventoried is called the source bucket, and the bucket where the inventory flat file is stored is called the destination bucket. The destination bucket must be in the same AWS Region as the source bucket.

When you configure an inventory for a source bucket, you specify the destination bucket where you want the inventory to be stored, and whether to generate the inventory daily or weekly. You can also configure what object metadata to include and whether to inventory all object versions or only current versions. For more information, see Amazon S3 Inventory in the Amazon Simple Storage Service Developer Guide.

Important
You must create a bucket policy on the destination bucket to grant permissions to Amazon S3 to write objects to the bucket in the defined location. For an example policy, see Granting Permissions for Amazon S3 Inventory and Storage Class Analysis.

To use this operation, you must have permissions to perform the s3:PutInventoryConfiguration action. The bucket owner has this permission by default and can grant this permission to others. For more information about permissions, see Permissions Related to Bucket Subresource Operations and Managing Access Permissions to Your Amazon S3 Resources in the Amazon Simple Storage Service Developer Guide.

Special Errors

- HTTP 400 Bad Request Error
  - Code: InvalidArgument
  - Cause: Invalid Argument
- HTTP 400 Bad Request Error
  - Code: TooManyConfigurations
  - Cause: You are attempting to create a new configuration but have already reached the 1,000-configuration limit.
- HTTP 403 Forbidden Error
  - Code: AccessDenied
  - Cause: You are not the owner of the specified bucket, or you do not have the s3:PutInventoryConfiguration bucket permission to set the configuration on the bucket.

Related Resources

- GetBucketInventoryConfiguration
- DeleteBucketInventoryConfiguration
- ListBucketInventoryConfigurations

Request Syntax

```
PUT //?inventory&id=Id HTTP/1.1
Host: Bucket.s3.amazonaws.com
x-amz-expected-bucket-owner: ExpectedBucketOwner
```
URI Request Parameters

The request uses the following URI parameters.

**Bucket (p. 281)**

The name of the bucket where the inventory configuration will be stored.

Required: Yes

**id (p. 281)**

The ID used to identify the inventory configuration.

Required: Yes

**x-amz-expected-bucket-owner (p. 281)**

The account id of the expected bucket owner. If the bucket is owned by a different account, the request will fail with an HTTP 403 (Access Denied) error.

Request Body

The request accepts the following data in XML format.

**InventoryConfiguration (p. 281)**

Root level tag for the InventoryConfiguration parameters.

Required: Yes

**Destination (p. 281)**

Contains information about where to publish the inventory results.
Type: InventoryDestination (p. 577) data type
Required: Yes

Filter (p. 281)
Specifies an inventory filter. The inventory only includes objects that meet the filter's criteria.
Type: InventoryFilter (p. 579) data type
Required: No

Id (p. 281)
The ID used to identify the inventory configuration.
Type: String
Required: Yes

IncludedObjectVersions (p. 281)
Object versions to include in the inventory list. If set to All, the list includes all the object versions, which adds the version-related fields VersionId, IsLatest, and DeleteMarker to the list. If set to Current, the list does not contain these version-related fields.
Type: String
Valid Values: All | Current
Required: Yes

 isEnabled (p. 281)
Specifies whether the inventory is enabled or disabled. If set to True, an inventory list is generated. If set to False, no inventory list is generated.
Type: Boolean
Required: Yes

OptionalFields (p. 281)
Contains the optional fields that are included in the inventory results.
Type: Array of strings
Valid Values: Size | LastModifiedDate | StorageClass | ETag |
IsMultipartUploaded | ReplicationStatus | EncryptionStatus |
ObjectLockRetainUntilDate | ObjectLockMode | ObjectLockLegalHoldStatus |
IntelligentTieringAccessTier
Required: No

Schedule (p. 281)
Specifies the schedule for generating inventory results.
Type: InventorySchedule (p. 582) data type
Required: Yes

Response Syntax
HTTP/1.1 200
Response Elements

If the action is successful, the service sends back an HTTP 200 response with an empty HTTP body.

Examples

Example: Create an inventory configuration

The following PUT request and response for the bucket examplebucket creates a new or replaces an existing inventory configuration with the ID report1. The configuration is defined in the request body.

```xml
<?xml version="1.0" encoding="UTF-8"?>
<InventoryConfiguration xmlns="http://s3.amazonaws.com/doc/2006-03-01/"
  <Id>report1</Id>
  <IsEnabled>true</IsEnabled>
  <Filter>
    <Prefix>filterPrefix</Prefix>
  </Filter>
  <Destination>
    <S3BucketDestination>
      <Format/csv</Format>
      <AccountId>123456789012</AccountId>
      <Bucket>arn:aws:s3:::destination-bucket</Bucket>
      <Prefix>prefix1</Prefix>
      <Encryption>
        <SSE-KMS>
          <KeyId>arn:aws:kms:us-west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab</KeyId>
        </SSE-KMS>
      </Encryption>
    </S3BucketDestination>
  </Destination>
  <Schedule>
    <Frequency>Daily</Frequency>
  </Schedule>
  <IncludedObjectVersions>All</IncludedObjectVersions>
  <OptionalFields>
    <Field>Size</Field>
    <Field>LastModifiedDate</Field>
    <Field>ETag</Field>
    <Field>StorageClass</Field>
    <Field>IsMultipartUploaded</Field>
    <Field>ReplicationStatus</Field>
    <Field>EncryptionStatus</Field>
    <Field>ObjectLockRetainUntilDate</Field>
    <Field>ObjectLockMode</Field>
    <Field>ObjectLockLegalHoldStatus</Field>
  </OptionalFields>
</InventoryConfiguration>
```

HTTP/1.1 200 OK
x-amz-id-2: YgIPfBiKa2bj0KMG95r/0zo3emzU4dzsD4rcKCHQUAQk3ShJTOOpXUueF6QKo
x-amz-request-id: 236A8905248E5A01
Date: Mon, 31 Oct 2016 12:00:00 GMT
See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3
PutBucketLifecycle
Service: Amazon Simple Storage Service

Important
For an updated version of this API, see PutBucketLifecycleConfiguration. This version has been deprecated. Existing lifecycle configurations will work. For new lifecycle configurations, use the updated API.

Creates a new lifecycle configuration for the bucket or replaces an existing lifecycle configuration. For information about lifecycle configuration, see Object Lifecycle Management in the Amazon Simple Storage Service Developer Guide.

By default, all Amazon S3 resources, including buckets, objects, and related subresources (for example, lifecycle configuration and website configuration) are private. Only the resource owner, the AWS account that created the resource, can access it. The resource owner can optionally grant access permissions to others by writing an access policy. For this operation, users must get the s3:PutLifecycleConfiguration permission.

You can also explicitly deny permissions. Explicit denial also supersedes any other permissions. If you want to prevent users or accounts from removing or deleting objects from your bucket, you must deny them permissions for the following actions:

- s3:DeleteObject
- s3:DeleteObjectVersion
- s3:PutLifecycleConfiguration

For more information about permissions, see Managing Access Permissions to your Amazon S3 Resources in the Amazon Simple Storage Service Developer Guide.

For more examples of transitioning objects to storage classes such as STANDARD_IA or ONEZONE_IA, see Examples of Lifecycle Configuration.

Related Resources
- GetBucketLifecycle(Deprecated)
- GetBucketLifecycleConfiguration
- RestoreObject
- By default, a resource owner—in this case, a bucket owner, which is the AWS account that created the bucket—can perform any of the operations. A resource owner can also grant others permission to perform the operation. For more information, see the following topics in the Amazon Simple Storage Service Developer Guide:
  - Specifying Permissions in a Policy
  - Managing Access Permissions to your Amazon S3 Resources

Request Syntax

```
PUT /?lifecycle HTTP/1.1
Host: Bucket.s3.amazonaws.com
Content-MD5: ContentMD5
x-amz-expected-bucket-owner: ExpectedBucketOwner
<?xml version="1.0" encoding="UTF-8"?>
<LifecycleConfiguration xmlns="http://s3.amazonaws.com/doc/2006-03-01/">
  <Rule>
    <AbortIncompleteMultipartUpload>
      <DaysAfterInitiation>integer</DaysAfterInitiation>
    </AbortIncompleteMultipartUpload>
  </Rule>
</LifecycleConfiguration>
```
<Expiration>
  <Days>integer</Days>
  <ExpiredObjectDeleteMarker>boolean</ExpiredObjectDeleteMarker>
</Expiration>

<ID>string</ID>

<NoncurrentVersionExpiration>
  <NoncurrentDays>integer</NoncurrentDays>
</NoncurrentVersionExpiration>

<NoncurrentVersionTransition>
  <NoncurrentDays>integer</NoncurrentDays>
  <StorageClass>string</StorageClass>
</NoncurrentVersionTransition>

<Prefix>string</Prefix>

<Status>string</Status>

<Transition>
  <Date>timestamp</Date>
  <Days>integer</Days>
  <StorageClass>string</StorageClass>
</Transition>

</Rule>

</LifecycleConfiguration>

URI Request Parameters

The request uses the following URI parameters.

**Bucket (p. 286)**

Required: Yes

**Content-MD5 (p. 286)**

*This header has been deprecated.*

**x-amz-expected-bucket-owner (p. 286)**

The account id of the expected bucket owner. If the bucket is owned by a different account, the request will fail with an HTTP 403 (Access Denied) error.

Request Body

The request accepts the following data in XML format.

**LifecycleConfiguration (p. 286)**

Root level tag for the LifecycleConfiguration parameters.

Required: Yes

**Rule (p. 286)**

Specifies lifecycle configuration rules for an Amazon S3 bucket.

Type: Array of Rule (p. 647) data types

Required: Yes

Response Syntax

```
HTTP/1.1 200
```
Response Elements

If the action is successful, the service sends back an HTTP 200 response with an empty HTTP body.

Examples

Sample Request: Body of a basic lifecycle configuration

In the request, you specify the lifecycle configuration in the request body. The lifecycle configuration is specified as XML. The following is an example of a basic lifecycle configuration. It specifies one rule. The Prefix in the rule identifies objects to which the rule applies. The rule also specifies two actions (Transition and Expiration). Each action specifies a timeline when Amazon S3 should perform the action. The Status indicates whether the rule is enabled or disabled.

```xml
<LifecycleConfiguration>
  <Rule>
    <ID>sample-rule</ID>
    <Prefix>key-prefix</Prefix>
    <Status>rule-status</Status>
    <Transition>
      <Date>value</Date>
      <StorageClass>storage class</StorageClass>
    </Transition>
    <Expiration>
      <Days>value</Days>
    </Expiration>
  </Rule>
</LifecycleConfiguration>
```

Sample Request: Body of a lifecycle configuration specifying noncurrent versions

If the state of your bucket is versioning-enabled or versioning-suspended, you can have many versions of the same object: one current version and zero or more noncurrent versions. The following lifecycle configuration specifies the actions (NoncurrentVersionTransition, NoncurrentVersionExpiration) that are specific to noncurrent object versions.

```xml
<LifecycleConfiguration>
  <Rule>
    <ID>sample-rule</ID>
    <Prefix>key-prefix</Prefix>
    <Status>rule-status</Status>
    <NoncurrentVersionTransition>
      <NoncurrentDays>value</NoncurrentDays>
      <StorageClass>storage class</StorageClass>
    </NoncurrentVersionTransition>
    <NoncurrentVersionExpiration>
      <NoncurrentDays>value</NoncurrentDays>
    </NoncurrentVersionExpiration>
  </Rule>
</LifecycleConfiguration>
```

Sample Request: Body of a lifecycle configuration that specifies a rule with AbortIncompleteMultipartUpload

You can use the multipart upload API to upload large objects in parts. For more information about multipart uploads, see Multipart Upload Overview in the Amazon Simple Storage Service Developer Guide. With lifecycle configuration, you can tell Amazon S3 to abort incomplete multipart uploads, which are
identified by the key name prefix specified in the rule, if they don't complete within a specified number of days. When Amazon S3 aborts a multipart upload, it deletes all parts associated with the upload. This ensures that you don't have incomplete multipart uploads that have left parts stored in Amazon S3, so you don't have to pay storage costs for them. The following is an example lifecycle configuration that specifies a rule with the AbortIncompleteMultipartUpload action. This action tells Amazon S3 to abort incomplete multipart uploads seven days after initiation.

```
<LifecycleConfiguration>
  <Rule>
    <ID>sample-rule</ID>
    <Prefix>SomeKeyPrefix</Prefix>
    <Status>rule-status</Status>
    <AbortIncompleteMultipartUpload>
      <DaysAfterInitiation>7</DaysAfterInitiation>
    </AbortIncompleteMultipartUpload>
  </Rule>
</LifecycleConfiguration>
```

Add lifecycle configuration to a bucket that is not versioning-enabled

The following is a sample `PUT /?lifecycle` request that adds the lifecycle configuration to the examplebucket bucket. The lifecycle configuration specifies two rules, each with one action:

- The Transition action tells Amazon S3 to transition objects with the "documents/" prefix to the GLACIER storage class 30 days after creation.
- The Expiration action tells Amazon S3 to delete objects with the "logs/" prefix 365 days after creation.

The sample response follows the sample request.

```
PUT /?lifecycle HTTP/1.1
Host: examplebucket.s3.<Region>.amazonaws.com
x-amz-date: Wed, 14 May 2014 02:11:21 GMT
Content-MD5: q6yJD1kCbaGGFb3QLY69A==
Authorization: authorization string
Content-Length: 415

<LifecycleConfiguration>
  <Rule>
    <ID>id1</ID>
    <Prefix>documents/</Prefix>
    <Status>Enabled</Status>
    <Transition>
      <Days>30</Days>
      <StorageClass>GLACIER</StorageClass>
    </Transition>
  </Rule>
  <Rule>
    <ID>id2</ID>
    <Prefix>logs/</Prefix>
    <Status>Enabled</Status>
    <Expiration>
      <Days>365</Days>
    </Expiration>
  </Rule>
</LifecycleConfiguration>
```

HTTP/1.1 200 OK
Add lifecycle configuration to a bucket that is versioning-enabled

The following is a sample PUT /?lifecycle request that adds the lifecycle configuration to the examplebucket bucket. The lifecycle configuration specifies two rules, each with one action. You specify these actions when your bucket is versioning-enabled or versioning is suspended:

- The **NoncurrentVersionExpiration** action tells Amazon S3 to expire noncurrent versions of objects with the “logs/” prefix 100 days after the objects become noncurrent.
- The **NoncurrentVersionTransition** action tells Amazon S3 to transition noncurrent versions of objects with the “documents/” prefix to the GLACIER storage class 30 days after they become noncurrent.

The sample response follows the sample request.

```xml
<LifecycleConfiguration>
  <Rule>
    <ID>id1</ID>
    <Prefix>logs/</Prefix>
    <Status>Enabled</Status>
    <NoncurrentVersionExpiration>
      <NoncurrentDays>1</NoncurrentDays>
    </NoncurrentVersionExpiration>
  </Rule>
  <Rule>
    <ID>TransitionSoonAfterBecomingNonCurrent</ID>
    <Prefix>documents/</Prefix>
    <Status>Enabled</Status>
    <NoncurrentVersionTransition>
      <NoncurrentDays>0</NoncurrentDays>
      <StorageClass>GLACIER</StorageClass>
    </NoncurrentVersionTransition>
  </Rule>
</LifecycleConfiguration>
```

**HTTP/1.1 200 OK**

```
x-amz-id-2: aXQ+sUpmWmO8//3hMdDTw/CnJArwje+J49Hf+J44yRb/Vmb1kgI05A+PT98cp/6k07hF+LD2mY=  
x-amz-request-id: 02D7EC4C10381EB1  
Date: Wed, 14 May 2014 02:21:50 GMT  
Content-Length: 0  
Server: AmazonS3
```

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:
• AWS Command Line Interface
• AWS SDK for .NET
• AWS SDK for C++
• AWS SDK for Go
• AWS SDK for Java
• AWS SDK for JavaScript
• AWS SDK for PHP V3
• AWS SDK for Python
• AWS SDK for Ruby V3
PutBucketLifecycleConfiguration

Service: Amazon Simple Storage Service

Creates a new lifecycle configuration for the bucket or replaces an existing lifecycle configuration. For information about lifecycle configuration, see Managing Access Permissions to Your Amazon S3 Resources.

**Note**

Bucket lifecycle configuration now supports specifying a lifecycle rule using an object key name prefix, one or more object tags, or a combination of both. Accordingly, this section describes the latest API. The previous version of the API supported filtering based only on an object key name prefix, which is supported for backward compatibility. For the related API description, see PutBucketLifecycle.

**Rules**

You specify the lifecycle configuration in your request body. The lifecycle configuration is specified as XML consisting of one or more rules. Each rule consists of the following:

- Filter identifying a subset of objects to which the rule applies. The filter can be based on a key name prefix, object tags, or a combination of both.
- Status whether the rule is in effect.
- One or more lifecycle transition and expiration actions that you want Amazon S3 to perform on the objects identified by the filter. If the state of your bucket is versioning-enabled or versioning-suspended, you can have many versions of the same object (one current version and zero or more noncurrent versions). Amazon S3 provides predefined actions that you can specify for current and noncurrent object versions.

For more information, see Object Lifecycle Management and Lifecycle Configuration Elements.

**Permissions**

By default, all Amazon S3 resources are private, including buckets, objects, and related subresources (for example, lifecycle configuration and website configuration). Only the resource owner (that is, the AWS account that created it) can access the resource. The resource owner can optionally grant access permissions to others by writing an access policy. For this operation, a user must get the s3:PutLifecycleConfiguration permission.

You can also explicitly deny permissions. Explicit deny also supersedes any other permissions. If you want to block users or accounts from removing or deleting objects from your bucket, you must deny them permissions for the following actions:

- s3:DeleteObject
- s3:DeleteObjectVersion
- s3:PutLifecycleConfiguration

For more information about permissions, see Managing Access Permissions to Your Amazon S3 Resources.

The following are related to PutBucketLifecycleConfiguration:

- Examples of Lifecycle Configuration
- GetBucketLifecycleConfiguration
- DeleteBucketLifecycle
### Request Syntax

```
PUT /?lifecycle HTTP/1.1
Host: Bucket.s3.amazonaws.com
x-amz-expected-bucket-owner: ExpectedBucketOwner
<?xml version="1.0" encoding="UTF-8"?>
<LifecycleConfiguration xmlns="http://s3.amazonaws.com/doc/2006-03-01/">
  <Rule>
    <AbortIncompleteMultipartUpload>
      <DaysAfterInitiation>integer</DaysAfterInitiation>
    </AbortIncompleteMultipartUpload>
    <Expiration>
      <Date>timestamp</Date>
      <Days>integer</Days>
      <ExpiredObjectDeleteMarker>boolean</ExpiredObjectDeleteMarker>
    </Expiration>
    <Filter>
      <And>
        <Prefix>string</Prefix>
        <Tag>
          <Key>string</Key>
          <Value>string</Value>
        </Tag>
        ...
      </And>
      <Prefix>string</Prefix>
      <Tag>
        <Key>string</Key>
        <Value>string</Value>
      </Tag>
    </Filter>
    <ID>string</ID>
    <NoncurrentVersionExpiration>
      <NoncurrentDays>integer</NoncurrentDays>
    </NoncurrentVersionExpiration>
    <NoncurrentVersionTransition>
      <NoncurrentDays>integer</NoncurrentDays>
      <StorageClass>string</StorageClass>
    </NoncurrentVersionTransition>
    ...
    <Prefix>string</Prefix>
    <Status>string</Status>
    <Transition>
      <Date>timestamp</Date>
      <Days>integer</Days>
      <StorageClass>string</StorageClass>
    </Transition>
    ...
  </Rule>
  ...
</LifecycleConfiguration>
```

### URI Request Parameters

The request uses the following URI parameters.

**Bucket (p. 293)**

The name of the bucket for which to set the configuration.

Required: Yes
x-amz-expected-bucket-owner (p. 293)

The account id of the expected bucket owner. If the bucket is owned by a different account, the request will fail with an HTTP 403 (Access Denied) error.

Request Body

The request accepts the following data in XML format.

LifecycleConfiguration (p. 293)

Root level tag for the LifecycleConfiguration parameters.

Required: Yes

Rule (p. 293)

A lifecycle rule for individual objects in an Amazon S3 bucket.

Type: Array of LifecycleRule (p. 589) data types

Required: Yes

Response Syntax

HTTP/1.1 200

Response Elements

If the action is successful, the service sends back an HTTP 200 response with an empty HTTP body.

Examples

Example 1: Add lifecycle configuration - bucket not versioning-enabled

The following lifecycle configuration specifies two rules, each with one action.

- The Transition action requests Amazon S3 to transition objects with the "documents/" prefix to the GLACIER storage class 30 days after creation.
- The Expiration action requests Amazon S3 to delete objects with the "logs/" prefix 365 days after creation.

```
<LifecycleConfiguration>
  <Rule>
    <ID>id1</ID>
    <Filter>
      <Prefix>documents/</Prefix>
    </Filter>
    <Status>Enabled</Status>
    <Transition>
      <Days>30</Days>
      <StorageClass>GLACIER</StorageClass>
    </Transition>
  </Rule>
  <Rule>
    <ID>id2</ID>
    <Filter>
      <Prefix>logs/</Prefix>
    </Filter>
    <Status>Enabled</Status>
    <Expiration>
      <Days>365</Days>
    </Expiration>
  </Rule>
</LifecycleConfiguration>
```
Example

The following is a sample PUT /?lifecycle request that adds the preceding lifecycle configuration to the examplebucket bucket.

```
PUT /?lifecycle HTTP/1.1
Host: examplebucket.s3.<Region>.amazonaws.com
x-amz-date: Wed, 14 May 2014 02:11:21 GMT
Content-MD5: q6yJDlIkcBaGGgf3QLY69A==
Authorization: authorization string
Content-Length: 415

<LifecycleConfiguration>
  <Rule>
    <ID>id1</ID>
    <Filter>
      <Prefix>documents/</Prefix>
    </Filter>
    <Status>Enabled</Status>
    <Transition>
      <Days>30</Days>
      <StorageClass>GLACIER</StorageClass>
    </Transition>
  </Rule>
  <Rule>
    <ID>id2</ID>
    <Filter>
      <Prefix>logs/</Prefix>
    </Filter>
    <Status>Enabled</Status>
    <Expiration>
      <Days>365</Days>
    </Expiration>
  </Rule>
</LifecycleConfiguration>
```

Sample Response

This example illustrates one usage of PutBucketLifecycleConfiguration.

```
HTTP/1.1 200 OK
x-amz-id-2: r+Q7+nhXtJDDJ30JJYcd+1j5nM/rUFiiiz/fNbdOsd3JUE8NWMLHNKhmmPfwMpd
x-amz-request-id: 9E26D00072A8EF9E
Date: Wed, 14 May 2014 02:11:22 GMT
Content-Length: 0
Server: AmazonS3
```
Example 2: Add lifecycle configuration - bucket is versioning-enabled

The following lifecycle configuration specifies two rules, each with one action for Amazon S3 to perform. You specify these actions when your bucket is versioning-enabled or versioning is suspended:

- The `NoncurrentVersionExpiration` action requests Amazon S3 to expire noncurrent versions of objects with the "logs/" prefix 100 days after the objects become noncurrent.
- The `NoncurrentVersionTransition` action requests Amazon S3 to transition noncurrent versions of objects with the "documents/" prefix to the GLACIER storage class 30 days after they become noncurrent.

```
<LifeCycleConfiguration>
    <Rule>
        <ID>DeleteAfterBecomingNonCurrent</ID>
        <Filter>
            <Prefix>logs/</Prefix>
        </Filter>
        <Status>Enabled</Status>
        <NoncurrentVersionExpiration>
            <NoncurrentDays>100</NoncurrentDays>
        </NoncurrentVersionExpiration>
    </Rule>
    <Rule>
        <ID>TransitionAfterBecomingNonCurrent</ID>
        <Filter>
            <Prefix>documents/</Prefix>
        </Filter>
        <Status>Enabled</Status>
        <NoncurrentVersionTransition>
            <NoncurrentDays>30</NoncurrentDays>
            <StorageClass>GLACIER</StorageClass>
        </NoncurrentVersionTransition>
    </Rule>
</LifeCycleConfiguration>
```

Example

The following is a sample PUT `/?lifecycle` request that adds the preceding lifecycle configuration to the examplebucket bucket.

```
PUT /?lifecycle HTTP/1.1
Host: examplebucket.s3.<Region>.amazonaws.com
x-amz-date: Wed, 14 May 2014 02:21:48 GMT
Content-MD5: 96rxH9mDqVNKkaZDddg==
Authorization: authorization string
Content-Length: 598

<LifeCycleConfiguration>
    <Rule>
        <ID>DeleteAfterBecomingNonCurrent</ID>
        <Filter>
            <Prefix>logs/</Prefix>
        </Filter>
        <Status>Enabled</Status>
        <NoncurrentVersionExpiration>
            <NoncurrentDays>1</NoncurrentDays>
        </NoncurrentVersionExpiration>
    </Rule>
</LifeCycleConfiguration>
```
Sample Response

This example illustrates one usage of PutBucketLifecycleConfiguration.

HTTP/1.1 200 OK
x-amz-id-2: aXQ+KbIrmMmoO//3bMdDTw/CnjArwje+J49Hf+j44yRb/VmbIkgIO5A+PT98Cp/6k07hf+LD2mY=
x-amz-request-id: 02D7EC4C10381EB1
Date: Wed, 14 May 2014 02:21:50 GMT
Content-Length: 0
Server: AmazonS3

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3
PutBucketLogging
Service: Amazon Simple Storage Service

Set the logging parameters for a bucket and to specify permissions for who can view and modify the logging parameters. All logs are saved to buckets in the same AWS Region as the source bucket. To set the logging status of a bucket, you must be the bucket owner.

The bucket owner is automatically granted FULL_CONTROL to all logs. You use the Grantee request element to grant access to other people. The Permissions request element specifies the kind of access the grantee has to the logs.

Grantee Values

You can specify the person (grantee) to whom you're assigning access rights (using request elements) in the following ways:

• By the person’s ID:

```
<Grantee xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:type="CanonicalUser"><ID><IDID></ID><DisplayName><GranteesEmail</DisplayName></Grantee>
```

DisplayNamewithinthe request. It is optional and ignored in the request.
• By Email address:

```
<Grantee xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:type="AmazonCustomerByEmail"><EmailAddress><Grantees@email.com></EmailAddress></Grantee>
```

The grantee is resolved to the CanonicalUser and, in a response to a GET Object acl request, appears as the CanonicalUser.
• By URI:

```
```

To enable logging, you use LoggingEnabled and its children request elements. To disable logging, you use an empty BucketLoggingStatus request element:

```
```

For more information about server access logging, see Server Access Logging.

For more information about creating a bucket, see CreateBucket. For more information about returning the logging status of a bucket, see GetBucketLogging.

The following operations are related to PutBucketLogging:

• PutObject
• DeleteBucket
• CreateBucket
• GetBucketLogging

Request Syntax

```
PUT //?logging HTTP/1.1
```
Host: Bucket.s3.amazonaws.com
Content-MD5: ContentMD5
x-amz-expected-bucket-owner: ExpectedBucketOwner

<?xml version="1.0" encoding="UTF-8"?>
  <LoggingEnabled>
    <TargetBucket>string</TargetBucket>
    <TargetGrants>
      <Grant>
        <Grantee>
          <DisplayName>string</DisplayName>
          <EmailAddress>string</EmailAddress>
          <ID>string</ID>
          <xsi:type>string</xsi:type>
          <URI>string</URI>
        </Grantee>
        <Permission>string</Permission>
      </Grant>
    </TargetGrants>
    <TargetPrefix>string</TargetPrefix>
  </LoggingEnabled>
</BucketLoggingStatus>

**URI Request Parameters**

The request uses the following URI parameters.

**Bucket (p. 298)**

The name of the bucket for which to set the logging parameters.

Required: Yes

**Content-MD5 (p. 298)**

*This header has been deprecated.*

The MD5 hash of the PutBucketLogging request body.

**x-amz-expected-bucket-owner (p. 298)**

The account id of the expected bucket owner. If the bucket is owned by a different account, the request will fail with an HTTP 403 (Access Denied) error.

**Request Body**

The request accepts the following data in XML format.

**BucketLoggingStatus (p. 298)**

Root level tag for the BucketLoggingStatus parameters.

Required: Yes

**LoggingEnabled (p. 298)**

Describes where logs are stored and the prefix that Amazon S3 assigns to all log object keys for a bucket. For more information, see PUT Bucket logging in the *Amazon Simple Storage Service API Reference*.

Type: LoggingEnabled (p. 593) data type

Required: No
Response Syntax

HTTP/1.1 200

Response Elements

If the action is successful, the service sends back an HTTP 200 response with an empty HTTP body.

Examples

Sample Request

This request enables logging and gives the grantee of the bucket READ access to the logs.

```
PUT ?logging HTTP/1.1
Host: quotes.s3.<Region>.amazonaws.com
Content-Length: 214
Date: Wed, 25 Nov 2009 12:00:00 GMT
Authorization: authorization string

<?xml version="1.0" encoding="UTF-8"?>
<BucketLoggingStatus xmlns="http://doc.s3.amazonaws.com/2006-03-01">
  <LoggingEnabled>
    <TargetBucket>mybucketlogs</TargetBucket>
    <TargetPrefix>mybucket-access_log-/</TargetPrefix>
    <TargetGrants>
      <Grant>
        <Grantee xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
          xsi:type="AmazonCustomerByEmail">
          <EmailAddress>user@company.com</EmailAddress>
        </Grantee>
        <Permission>READ</Permission>
      </Grant>
    </TargetGrants>
  </LoggingEnabled>
</BucketLoggingStatus>
```

Sample Response

This example illustrates one usage of PutBucketLogging.

```
HTTP/1.1 200 OK
x-amz-id-2: YgIPIfBiKa2bj0KMg95r/0zo3emsU4dzzsD4rcKCHQUAdQkfJ3ShJTOOpXUueF6QKo
x-amz-request-id: 236A8905246E5A01
Date: Wed, 01 Mar 2006 12:00:00 GMT
```

Sample Request: Disabling logging

This request disables logging on the bucket quotes.

```
PUT ?logging HTTP/1.1
Host: quotes.s3.<Region>.amazonaws.com
Content-Length: 214
Date: Wed, 25 Nov 2009 12:00:00 GMT
```
Authorization: authorization string

```xml
<?xml version="1.0" encoding="UTF-8"?>
```

**Sample Response**

This example illustrates one usage of `PutBucketLogging`.

```
HTTP/1.1 200 OK
x-amz-id-2: YgIPIfBiKa2bji0km95r/0zo3emzU4dzsD4rcKCHquADqk6fS3WhJTOOpXUueF6QKo
x-amz-request-id: 236A8905248E5A01
Date: Wed, 01 Mar 2006 12:00:00 GMT
```

**See Also**

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3
**PutBucketMetricsConfiguration**

Service: Amazon Simple Storage Service

Sets a metrics configuration (specified by the metrics configuration ID) for the bucket. You can have up to 1,000 metrics configurations per bucket. If you’re updating an existing metrics configuration, note that this is a full replacement of the existing metrics configuration. If you don’t include the elements you want to keep, they are erased.

To use this operation, you must have permissions to perform the `s3:PutMetricsConfiguration` action. The bucket owner has this permission by default. The bucket owner can grant this permission to others. For more information about permissions, see Permissions Related to Bucket Subresource Operations and Managing Access Permissions to Your Amazon S3 Resources.

For information about CloudWatch request metrics for Amazon S3, see Monitoring Metrics with Amazon CloudWatch.

The following operations are related to `PutBucketMetricsConfiguration`:

- `DeleteBucketMetricsConfiguration`
- `PutBucketMetricsConfiguration`
- `ListBucketMetricsConfigurations`

GetBucketLifecycle has the following special error:

- Error code: `TooManyConfigurations`
  - Description: You are attempting to create a new configuration but have already reached the 1,000-configuration limit.
  - HTTP Status Code: HTTP 400 Bad Request

### Request Syntax

```xml
PUT /?metrics&id=Id HTTP/1.1
Host: Bucket.s3.amazonaws.com
x-amz-expected-bucket-owner: ExpectedBucketOwner
<?xml version="1.0" encoding="UTF-8"?>
  <Id>string</Id>
  <Filter>
    <And>
      <Prefix>string</Prefix>
      <Tag>
        <Key>string</Key>
        <Value>string</Value>
      </Tag>
      ...
    </And>
    <Prefix>string</Prefix>
    <Tag>
      <Key>string</Key>
      <Value>string</Value>
    </Tag>
  </Filter>
</MetricsConfiguration>
```

### URI Request Parameters

The request uses the following URI parameters.
**Bucket (p. 302)**

The name of the bucket for which the metrics configuration is set.

**Required:** Yes

**id (p. 302)**

The ID used to identify the metrics configuration.

**Required:** Yes

**x-amz-expected-bucket-owner (p. 302)**

The account id of the expected bucket owner. If the bucket is owned by a different account, the request will fail with an HTTP 403 (Access Denied) error.

**Request Body**

The request accepts the following data in XML format.

**MetricsConfiguration (p. 302)**

Root level tag for the MetricsConfiguration parameters.

**Required:** Yes

**Filter (p. 302)**

Specifies a metrics configuration filter. The metrics configuration will only include objects that meet the filter's criteria. A filter must be a prefix, a tag, or a conjunction (MetricsAndOperator).

**Type:** MetricsFilter (p. 598) data type

**Required:** No

**Id (p. 302)**

The ID used to identify the metrics configuration.

**Type:** String

**Required:** Yes

**Response Syntax**

HTTP/1.1 200

**Response Elements**

If the action is successful, the service sends back an HTTP 200 response with an empty HTTP body.

**Examples**

**First Sample Request**

Put a metric configuration that enables metrics for an entire bucket.

```
PUT /?metrics&id=EntireBucket HTTP/1.1
```
**First Sample Response**

This example illustrates one usage of `PutBucketMetricsConfiguration`.

```
HTTP/1.1 204 No Content
x-amz-id-2: ITnGT1y4REXAMPLEPi4hklTXouTf0hccUjo0iCPEXAMPLEutBj3M7fPGLWO2SEWp
x-amz-request-id: 51991EXAMPLE5321
Date: Thu, 15 Nov 2016 00:17:22 GMT
Server: AmazonS3
```

**Second Sample Request**

Put a metrics configuration that enables metrics for objects that start with a particular prefix and also have specific tags applied.

```
PUT /?metrics&Id=ImportantBlueDocuments HTTP/1.1
Host: examplebucket.s3.<Region>.amazonaws.com
x-amz-date: Thu, 15 Nov 2016 00:17:29 GMT
Authorization: signatureValue
Content-Length: 480

<?xml version="1.0" encoding="UTF-8"?><MetricsConfiguration xmlns="http://s3.amazonaws.com/doc/2006-03-01/">
    <Id>ImportantBlueDocuments</Id>
    <Filter>
        <And>
            <Prefix>documents/</Prefix>
            <Tag>
                <Key>priority</Key>
                <Value>high</Value>
            </Tag>
            <Tag>
                <Key>class</Key>
                <Value>blue</Value>
            </Tag>
        </And>
    </Filter>
</MetricsConfiguration>
```

**Second Sample Response**

This example illustrates one usage of `PutBucketMetricsConfiguration`.

```
HTTP/1.1 204 No Content
x-amz-id-2: ITnGT1y4REXAMPLEPi4hklTXouTf0hccUjo0iCPEXAMPLEutBj3M7fPGLWO2SEWp
```

API Version 2006-03-01
See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3
PutBucketNotification
Service: Amazon Simple Storage Service

No longer used, see the PutBucketNotificationConfiguration operation.

Request Syntax

```
PUT /?notification HTTP/1.1
Host: Bucket.s3.amazonaws.com
Content-MD5: ContentMD5
x-amz-expected-bucket-owner: ExpectedBucketOwner
<?xml version="1.0" encoding="UTF-8"?>
  <TopicConfiguration>
    <Event>string</Event>
    <Event>string</Event>
    ...
    <Id>string</Id>
    <Topic>string</Topic>
  </TopicConfiguration>
  <QueueConfiguration>
    <Event>string</Event>
    <Event>string</Event>
    ...
    <Id>string</Id>
    <Queue>string</Queue>
  </QueueConfiguration>
  <CloudFunctionConfiguration>
    <CloudFunction>string</CloudFunction>
    <Event>string</Event>
    <Event>string</Event>
    ...
    <Id>string</Id>
    <InvocationRole>string</InvocationRole>
  </CloudFunctionConfiguration>
</NotificationConfiguration>
```

URI Request Parameters

The request uses the following URI parameters.

**Bucket** *(p. 306)*

The name of the bucket.

Required: Yes

**Content-MD5** *(p. 306)*

This header has been deprecated.

The MD5 hash of the PutPublicAccessBlock request body.

**x-amz-expected-bucket-owner** *(p. 306)*

The account id of the expected bucket owner. If the bucket is owned by a different account, the request will fail with an HTTP 403 (Access Denied) error.

Request Body

The request accepts the following data in XML format.
NotificationConfiguration (p. 306)

Root level tag for the NotificationConfiguration parameters.

Required: Yes

CloudFunctionConfiguration (p. 306)

Container for specifying the AWS Lambda notification configuration.

Type: CloudFunctionConfiguration (p. 526) data type

Required: No

QueueConfiguration (p. 306)

This data type is deprecated. This data type specifies the configuration for publishing messages to an Amazon Simple Queue Service (Amazon SQS) queue when Amazon S3 detects specified events.

Type: QueueConfigurationDeprecated (p. 629) data type

Required: No

TopicConfiguration (p. 306)

This data type is deprecated. A container for specifying the configuration for publication of messages to an Amazon Simple Notification Service (Amazon SNS) topic when Amazon S3 detects specified events.

Type: TopicConfigurationDeprecated (p. 672) data type

Required: No

Response Syntax

```
HTTP/1.1 200
```

Response Elements

If the action is successful, the service sends back an HTTP 200 response with an empty HTTP body.

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3
PutBucketNotificationConfiguration
Service: Amazon Simple Storage Service

Enables notifications of specified events for a bucket. For more information about event notifications, see Configuring Event Notifications.

Using this API, you can replace an existing notification configuration. The configuration is an XML file that defines the event types that you want Amazon S3 to publish and the destination where you want Amazon S3 to publish an event notification when it detects an event of the specified type.

By default, your bucket has no event notifications configured. That is, the notification configuration will be an empty NotificationConfiguration.

</NotificationConfiguration>

This operation replaces the existing notification configuration with the configuration you include in the request body.

After Amazon S3 receives this request, it first verifies that any Amazon Simple Notification Service (Amazon SNS) or Amazon Simple Queue Service (Amazon SQS) destination exists, and that the bucket owner has permission to publish to it by sending a test notification. In the case of AWS Lambda destinations, Amazon S3 verifies that the Lambda function permissions grant Amazon S3 permission to invoke the function from the Amazon S3 bucket. For more information, see Configuring Notifications for Amazon S3 Events.

You can disable notifications by adding the empty NotificationConfiguration element.

By default, only the bucket owner can configure notifications on a bucket. However, bucket owners can use a bucket policy to grant permission to other users to set this configuration with s3:PutBucketNotification permission.

Note
The PUT notification is an atomic operation. For example, suppose your notification configuration includes SNS topic, SQS queue, and Lambda function configurations. When you send a PUT request with this configuration, Amazon S3 sends test messages to your SNS topic. If the message fails, the entire PUT operation will fail, and Amazon S3 will not add the configuration to your bucket.

Responses

If the configuration in the request body includes only one TopicConfiguration specifying only the s3:ReducedRedundancyLostObject event type, the response will also include the x-amz-sns-test-message-id header containing the message ID of the test notification sent to the topic.

The following operation is related to PutBucketNotificationConfiguration:

• GetBucketNotificationConfiguration

Request Syntax

PUT /?notification HTTP/1.1
Host: Bucket.s3.amazonaws.com
x-amz-expected-bucket-owner: ExpectedBucketOwner
<?xml version="1.0" encoding="UTF-8"?><NotificationConfiguration xmlns="http://s3.amazonaws.com/doc/2006-03-01/"
  <TopicConfiguration>
    <Event>string</Event>
    ...
  <Filter>
URI Request Parameters

The request uses the following URI parameters.

**Bucket (p. 308)**

The name of the bucket.

Required: Yes

**x-amz-expected-bucket-owner (p. 308)**

The account id of the expected bucket owner. If the bucket is owned by a different account, the request will fail with an HTTP 403 (Access Denied) error.

Request Body

The request accepts the following data in XML format.
NotificationConfiguration (p. 308)

Root level tag for the NotificationConfiguration parameters.

Required: Yes

CloudFunctionConfiguration (p. 308)

Describes the AWS Lambda functions to invoke and the events for which to invoke them.

Type: Array of LambdaFunctionConfiguration (p. 585) data types

Required: No

QueueConfiguration (p. 308)

The Amazon Simple Queue Service queues to publish messages to and the events for which to publish messages.

Type: Array of QueueConfiguration (p. 627) data types

Required: No

TopicConfiguration (p. 308)

The topic to which notifications are sent and the events for which notifications are generated.

Type: Array of TopicConfiguration (p. 670) data types

Required: No

Response Syntax

HTTP/1.1 200

Response Elements

If the action is successful, the service sends back an HTTP 200 response with an empty HTTP body.

Examples

Example 1: Configure notification to invoke a cloud function in Lambda

The following notification configuration includes CloudFunctionConfiguration, which identifies the event type for which Amazon S3 can invoke a cloud function and the name of the cloud function to invoke.

```
<NotificationConfiguration>
  <CloudFunctionConfiguration>
    <Id>ObjectCreatedEvents</Id>
    <CloudFunction>arn:aws:lambda:us-west-2:35667example:function:CreateThumbnail</CloudFunction>
    <Event>s3:ObjectCreated:*</Event>
  </CloudFunctionConfiguration>
</NotificationConfiguration>
```

Example

The following PUT uploads the notification configuration. The operation replaces the existing notification configuration.
Sample Response

This example illustrates one usage of PutBucketNotificationConfiguration.

```
HTTP/1.1 200 OK
x-amz-id-2: 8+FlwagBSoT2qpMaGlCFUKrkFR5W30eS7UhhoBb17j+kqvPS2cSF1gJ5coLd53d2
x-amz-request-id: E5BA4600A3937335
Date: Fri, 31 Oct 2014 01:49:50 GMT
Content-Length: 0
Server: AmazonS3
```

Example 2: Configure a notification with multiple destinations

The following notification configuration includes the topic and queue configurations:

- A topic configuration identifying an SNS topic for Amazon S3 to publish events of the `s3:ReducedRedundancyLostObject` type.
- A queue configuration identifying an SQS queue for Amazon S3 to publish events of the `s3:ObjectCreated:*` type.

```
<NotificationConfiguration>
  <TopicConfiguration>
    <Topic>arn:aws:sns:us-east-1:356671443308:s3notificationtopic2</Topic>
    <Event>s3:ReducedRedundancyLostObject</Event>
  </TopicConfiguration>
  <QueueConfiguration>
    <Queue>arn:aws:sqs:us-east-1:356671443308:s3notificationqueue</Queue>
    <Event>s3:ObjectCreated:*</Event>
  </QueueConfiguration>
</NotificationConfiguration>
```

Example

The following PUT request against the notification subresource of the examplebucket bucket sends the preceding notification configuration in the request body. The operation replaces the existing notification configuration on the bucket.
Example 3: Configure a notification with object key name filtering

The following notification configuration contains a queue configuration identifying an Amazon SQS queue for Amazon S3 to publish events to of the s3:ObjectCreated:Put type. The events will be published whenever an object that has a prefix of images/ and a .jpg suffix is PUT to a bucket. For more examples of notification configurations that use filtering, see Configuring Event Notifications.

```
<NotificationConfiguration>
  <QueueConfiguration>
    <Id>1</Id>
    <Filter>
      <S3Key><FilterRule>
        <Name>prefix</Name>
        <Value>images/</Value>
      </FilterRule>
      <FilterRule>
        <Name>suffix</Name>
        <Value>.jpg</Value>
      </FilterRule>
    </S3Key>
  </Filter>
  <Queue>arn:aws:sqs:us-west-2:444455556666:s3notificationqueue</Queue>
  <Event>s3:ObjectCreated:Put</Event>
</QueueConfiguration>
</NotificationConfiguration>
```

Example

The following PUT request against the notification subresource of the examplebucket bucket sends the preceding notification configuration in the request body. The operation replaces the existing notification configuration on the bucket.

```
PUT http://s3.<Region>.amazonaws.com/examplebucket?notification= HTTP/1.1
User-Agent: s3curl 2.0
Host: s3.amazonaws.com
Pragma: no-cache
Accept: */*
Proxy-Connection: Keep-Alive
Authorization: authorization string
Date: Mon, 13 Oct 2014 22:58:43 +0000
Content-Length: length
Expect: 100-continue
```
**Sample Response**

This example illustrates one usage of `PutBucketNotificationConfiguration`.

```plaintext
HTTP/1.1 200 OK
x-amz-id-2: SlvJLkfunoAGILZK3KgHSSUq4kwbudkrROMEsoNHopDacULy+cxRoR18vfoyvg2A
x-amz-request-id: BB1BA8E12D5A0B7
Date: Mon, 13 Oct 2014 22:58:44 GMT
Content-Length: 0
Server: AmazonS3
```

**See Also**

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3
**PutBucketOwnershipControls**  
Service: Amazon Simple Storage Service

Creates or modifies OwnershipControls for an Amazon S3 bucket. To use this operation, you must have the s3:PutBucketOwnershipControls permission. For more information about Amazon S3 permissions, see Specifying Permissions in a Policy.

For information about Amazon S3 Object Ownership, see Using Object Ownership.

The following operations are related to PutBucketOwnershipControls:

- GetBucketOwnershipControls (p. 131)
- DeleteBucketOwnershipControls (p. 59)

**Request Syntax**

```xml
PUT /?ownershipControls HTTP/1.1
Host: Bucket.s3.amazonaws.com
Content-MD5: ContentMD5
x-amz-expected-bucket-owner: ExpectedBucketOwner
<?xml version="1.0" encoding="UTF-8"?>
<OwnershipControls xmlns="http://s3.amazonaws.com/doc/2006-03-01/">
  <Rule>
    <ObjectOwnership>string</ObjectOwnership>
  </Rule>
  ...
</OwnershipControls>
```

**URI Request Parameters**

The request uses the following URI parameters.

**Bucket (p. 314)**

The name of the Amazon S3 bucket whose OwnershipControls you want to set.

Required: Yes

**Content-MD5 (p. 314)**

The MD5 hash of the OwnershipControls request body.

**x-amz-expected-bucket-owner (p. 314)**

The account id of the expected bucket owner. If the bucket is owned by a different account, the request will fail with an HTTP 403 (Access Denied) error.

**Request Body**

The request accepts the following data in XML format.

**OwnershipControls (p. 314)**

Root level tag for the OwnershipControls parameters.

Required: Yes

**Rule (p. 314)**

The container element for an ownership control rule.
Type: Array of OwnershipControlsRule (p. 619) data types

Required: Yes

Response Syntax

```
HTTP/1.1 200
```

Response Elements

If the action is successful, the service sends back an HTTP 200 response with an empty HTTP body.

Examples

Sample Request with BucketOwnerPreferred OwnershipControls

The following request puts a bucket OwnershipControls that specifies BucketOwnerPreferred.

```
PUT /example-bucket?ownershipControls= HTTP/1.1
Host:example-bucket.s3.<Region>.amazonaws.com
x-amz-date: 20200618T230132Z
x-amz-content-sha256: bafb46c18574a73704c8227aef060df1c12ea0d964e19b949d06e9f763805fe2
Authorization: authorization string

<?xml version="1.0" encoding="UTF-8"?>
<OwnershipControls xmlns="http://s3.amazonaws.com/doc/2006-03-01/">
  <Rule>
    <ObjectOwnership>BucketOwnerPreferred</ObjectOwnership>
  </Rule>
</OwnershipControls>
```

Sample Response with BucketOwnerPreferred OwnershipControls

This example illustrates one usage of PutBucketOwnershipControls.

```
HTTP/1.1 200 OK
x-amz-id-2: zkDVX0gbzl0oKcjNz7GPz8xXkhNArHta8/WDofshyEj6SbissRdqIT3vSuAmik7HK4PY+1zDZzIO=
x-amz-request-id: BK7Y8MSG7ZORFRCP
Date: Thu, 18 Jun 2020 23:01:33 GMT
Content-Length: 0
Server: AmazonS3
```

Sample Request with ObjectWriter for BucketOwnershipControls

The following request puts a bucket OwnershipControls that specifies ObjectWriter.

```
PUT /example-bucket?ownershipControls= HTTP/1.1
Host:example-bucket.s3.<Region>.amazonaws.com
x-amz-date: 20200618T230132Z
x-amz-content-sha256: bafb46c18574a73704c8227aef060df1c12ea0d964e19b949d06e9f763805fe2
```

API Version 2006-03-01
Authorization: authorization string

```xml
<?xml version="1.0" encoding="UTF-8"?>
<OwnershipControls xmlns="http://s3.amazonaws.com/doc/2006-03-01/">
  <Rule>
    <ObjectOwnership>ObjectWriter</ObjectOwnership>
  </Rule>
</OwnershipControls>
```

**Sample Response with ObjectWriter for BucketOwnershipControls**

This example illustrates one usage of PutBucketOwnershipControls.

```
HTTP/1.1 200 OK
x-amz-id-2: zkDVX0gbz8oKcjNz7GPz8XhXkhNArHtA8/WOf5hyEj6SbissRdqITZvSuAMik7HK4FY
+izDZzIO=
x-amz-request-id: BK7Y8M3G7Z0RFRCP
Date: Thu, 18 Jun 2020 23:01:33 GMT
Content-Length: 0
Server: AmazonS3
```

**See Also**

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3
PutBucketPolicy
Service: Amazon Simple Storage Service

Applies an Amazon S3 bucket policy to an Amazon S3 bucket. If you are using an identity other than the root user of the AWS account that owns the bucket, the calling identity must have the PutBucketPolicy permissions on the specified bucket and belong to the bucket owner's account in order to use this operation.

If you don't have PutBucketPolicy permissions, Amazon S3 returns a 403 Access Denied error. If you have the correct permissions, but you're not using an identity that belongs to the bucket owner's account, Amazon S3 returns a 405 Method Not Allowed error.

Important
As a security precaution, the root user of the AWS account that owns a bucket can always use this operation, even if the policy explicitly denies the root user the ability to perform this action.

For more information about bucket policies, see Using Bucket Policies and User Policies.

The following operations are related to PutBucketPolicy:

- CreateBucket
- DeleteBucket

Request Syntax

```
PUT /?policy HTTP/1.1
Host: Bucket.s3.amazonaws.com
Content-MD5: ContentMD5
x-amz-confirm-remove-self-bucket-access: ConfirmRemoveSelfBucketAccess
x-amz-expected-bucket-owner: ExpectedBucketOwner

{ Policy in JSON format }
```

URI Request Parameters

The request uses the following URI parameters.

Bucket (p. 317)

The name of the bucket.

Required: Yes

Content-MD5 (p. 317)

This header has been deprecated.

The MD5 hash of the request body.

x-amz-confirm-remove-self-bucket-access (p. 317)

Set this parameter to true to confirm that you want to remove your permissions to change this bucket policy in the future.

x-amz-expected-bucket-owner (p. 317)

The account id of the expected bucket owner. If the bucket is owned by a different account, the request will fail with an HTTP 403 (Access Denied) error.
Request Body

The request accepts the following data in JSON format.

Policy (p. 317)

Response Syntax

```
HTTP/1.1 200
```

Response Elements

If the action is successful, the service sends back an HTTP 200 response with an empty HTTP body.

Examples

Sample Request

The following request shows the PUT individual policy request for the bucket.

```
PUT /?policy HTTP/1.1
Host: bucket.s3.<Region>.amazonaws.com
Date: Tue, 04 Apr 2010 20:34:56 GMT
Authorization: authorization string

{
  "Version":"2008-10-17",
  "Id":"aaaa-bbbb-cccc-dddd",
  "Statement" : [
    {
      "Effect":"Allow",
      "Sid":"1",
      "Principal" : {
        "AWS": ["111122223333","444455556666"]
      },
      "Action" : ["s3:*"],
      "Resource" : "arn:aws:s3:::bucket/*"
    }
  ]
}
```

Sample Response

This example illustrates one usage of PutBucketPolicy.

```
HTTP/1.1 204 No Content
x-amz-id-2: Uuag1LuByR5Onimru9SAMPLEAtRPfTaGFg==
x-amz-request-id: 656c76696e672773273734
Date: Tue, 04 Apr 2010 20:34:56 GMT
Connection: keep-alive
Server: AmazonS3
```

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:
• AWS Command Line Interface
• AWS SDK for .NET
• AWS SDK for C++
• AWS SDK for Go
• AWS SDK for Java
• AWS SDK for JavaScript
• AWS SDK for PHP V3
• AWS SDK for Python
• AWS SDK for Ruby V3
PutBucketReplication
Service: Amazon Simple Storage Service

Creates a replication configuration or replaces an existing one. For more information, see Replication in the Amazon S3 Developer Guide.

Note
To perform this operation, the user or role performing the operation must have the iam:PassRole permission.

Specify the replication configuration in the request body. In the replication configuration, you provide the name of the destination bucket where you want Amazon S3 to replicate objects, the IAM role that Amazon S3 can assume to replicate objects on your behalf, and other relevant information.

A replication configuration must include at least one rule, and can contain a maximum of 1,000. Each rule identifies a subset of objects to replicate by filtering the objects in the source bucket. To choose additional subsets of objects to replicate, add a rule for each subset. All rules must specify the same destination bucket.

To specify a subset of the objects in the source bucket to apply a replication rule to, add the Filter element as a child of the Rule element. You can filter objects based on an object key prefix, one or more object tags, or both. When you add the Filter element in the configuration, you must also add the following elements: DeleteMarkerReplication, Status, and Priority.

Note
If you are using an earlier version of the replication configuration, Amazon S3 handles replication of delete markers differently. For more information, see Backward Compatibility.

For information about enabling versioning on a bucket, see Using Versioning.

By default, a resource owner, in this case the AWS account that created the bucket, can perform this operation. The resource owner can also grant others permissions to perform the operation. For more information about permissions, see Specifying Permissions in a Policy and Managing Access Permissions to Your Amazon S3 Resources.

Handling Replication of Encrypted Objects

By default, Amazon S3 doesn't replicate objects that are stored at rest using server-side encryption with CMKs stored in AWS KMS. To replicate AWS KMS-encrypted objects, add the following: SourceSelectionCriteria, SseKmsEncryptedObjects, Status, EncryptionConfiguration, and ReplicaKmsKeyID. For information about replication configuration, see Replicating Objects Created with SSE Using CMKs stored in AWS KMS.

For information on PutBucketReplication errors, see List of replication-related error codes

The following operations are related to PutBucketReplication:

- GetBucketReplication
- DeleteBucketReplication

Request Syntax

```
PUT /?replication HTTP/1.1
Host: Bucket.s3.amazonaws.com
Content-MD5: ContentMD5
x-amz-bucket-object-lock-token: Token
x-amz-expected-bucket-owner: ExpectedBucketOwner
<?xml version="1.0" encoding="UTF-8"?>
  <Role>string</Role>
</ReplicationConfiguration>
```
<Rule>
  <DeleteMarkerReplication>
    <Status>string</Status>
  </DeleteMarkerReplication>
  <Destination>
    <AccessControlTranslation>
      <Owner>string</Owner>
    </AccessControlTranslation>
    <Account>string</Account>
    <Bucket>string</Bucket>
    <EncryptionConfiguration>
      <ReplicaKmsKeyID>string</ReplicaKmsKeyID>
    </EncryptionConfiguration>
    <Metrics>
      <EventThreshold>
        <Minutes>integer</Minutes>
      </EventThreshold>
      <Status>string</Status>
    </Metrics>
    <ReplicationTime>
      <Status>string</Status>
      <Time>
        <Minutes>integer</Minutes>
      </Time>
    </ReplicationTime>
    <StorageClass>string</StorageClass>
  </Destination>
  <ExistingObjectReplication>
    <Status>string</Status>
  </ExistingObjectReplication>
  <Filter>
    <And>
      <Prefix>string</Prefix>
      <Tag>
        <Key>string</Key>
        <Value>string</Value>
      </Tag>
    </And>
    <Prefix>string</Prefix>
    <Tag>
      <Key>string</Key>
      <Value>string</Value>
    </Tag>
  </Filter>
  <ID>string</ID>
  <Prefix>string</Prefix>
  <Priority>integer</Priority>
  <SourceSelectionCriteria>
    <SseKmsEncryptedObjects>
      <Status>string</Status>
    </SseKmsEncryptedObjects>
  </SourceSelectionCriteria>
  <Status>string</Status>
</Rule>

URI Request Parameters

The request uses the following URI parameters.

Bucket (p. 320)

The name of the bucket
Required: Yes

**Content-MD5 (p. 320)**

_This header has been deprecated._

The base64-encoded 128-bit MD5 digest of the data. You must use this header as a message integrity check to verify that the request body was not corrupted in transit. For more information, see RFC 1864.

**x-amz-bucket-object-lock-token (p. 320)**

A token to allow Object Lock to be enabled for an existing bucket.

**x-amz-expected-bucket-owner (p. 320)**

The account id of the expected bucket owner. If the bucket is owned by a different account, the request will fail with an HTTP 403 (Access Denied) error.

**Request Body**

The request accepts the following data in XML format.

**ReplicationConfiguration (p. 320)**

Root level tag for the ReplicationConfiguration parameters.

Required: Yes

**Role (p. 320)**

The Amazon Resource Name (ARN) of the AWS Identity and Access Management (IAM) role that Amazon S3 assumes when replicating objects. For more information, see How to Set Up Replication in the Amazon Simple Storage Service Developer Guide.

Type: String

Required: Yes

**Rule (p. 320)**

A container for one or more replication rules. A replication configuration must have at least one rule and can contain a maximum of 1,000 rules.

Type: Array of ReplicationRule (p. 636) data types

Required: Yes

**Response Syntax**

| HTTP/1.1 200 |

**Response Elements**

If the action is successful, the service sends back an HTTP 200 response with an empty HTTP body.

**Examples**

**Sample Request: Add a replication configuration**

The following is a sample PUT request that creates a replication subresource on the specified bucket and saves the replication configuration in it. The replication configuration specifies a rule to replicate objects.
to the exampletargetbucket bucket. The rule includes a filter to replicate only the objects created with the key name prefix TaxDocs and that have two specific tags.

After you add a replication configuration to your bucket, Amazon S3 assumes the AWS Identity and Access Management (IAM) role specified in the configuration to replicate objects on behalf of the bucket owner. The bucket owner is the AWS account that created the bucket.

Filtering using the <Filter> element is supported in the latest XML configuration. If you are using an earlier version of the XML configuration, you can filter only on key prefix. In that case, you add the <Prefix> element as a child of the <Rule>.

For more examples of replication configuration, see Replication Configuration Overview in the Amazon S3 Developer Guide.

```
PUT /?replication HTTP/1.1
Host: examplebucket.s3.<Region>.amazonaws.com
Date: Wed, 11 Feb 2015 02:11:21 GMT
Content-MD5: q6yJDlkCBA06fb3QLY69A==
Authorization: authorization string
Content-Length: length

<ReplicationConfiguration>
  <Role>arn:aws:iam::35667example:role/CrossRegionReplicationRoleForS3</Role>
  <Rule>
    <ID>rule1</ID>
    <Status>Enabled</Status>
    <Priority>1</Priority>
    <DeleteMarkerReplication>
      <Status>Disabled</Status>
    </DeleteMarkerReplication>
    <Filter>
      <And>
        <Prefix>TaxDocs</Prefix>
        <Tag>
          <Key>key1</Key>
          <Value>value1</Value>
        </Tag>
        <Tag>
          <Key>key1</Key>
          <Value>value1</Value>
        </Tag>
      </And>
    </Filter>
    <Destination>
      <Bucket>arn:aws:s3:::exampletargetbucket</Bucket>
    </Destination>
  </Rule>
</ReplicationConfiguration>
```

Sample Response

This example illustrates one usage of PutBucketReplication.

```
HTTP/1.1 200 OK
x-amz-id-2: r+qR7+nhXtJDDJ0JYcd+1j5nM/rUfiiz/fNbdOsd3JUE8NWMLNHKmPwPwMpc
x-amz-request-id: 9E26D08072A8EF9E
Date: Wed, 11 Feb 2015 02:11:22 GMT
Content-Length: 0
Server: AmazonS3
```
Sample Request: Add a Replication Configuration with Amazon S3 Replication Time Control Enabled

You can use S3 Replication Time Control (S3 RTC) to replicate your data in the same AWS Region or across different AWS Regions in a predictable time frame. S3 RTC replicates 99.99 percent of new objects stored in Amazon S3 within 15 minutes. For more information, see Replicating Objects Using Replication Time Control.

PUT /?replication HTTP/1.1
Host: examplebucket.s3.<Region>.amazonaws.com
Date: Wed, 11 Feb 2015 02:11:21 GMT
Content-MD5: q6yJDlkcBaGGfb3QLY69A==
Authorization: authorization string
Content-Length: length
x-amz-bucket-object-lock-token: Token
<?xml version="1.0" encoding="UTF-8"?>
<ReplicationConfiguration>
  <Role>arn:aws:iam::35667example:role/CrossRegionReplicationRoleForS3</Role>
  <Rule>
    <ID>rule1</ID>
    <Status>Enabled</Status>
    <Priority>1</Priority>
    <Filter>
      <And>
        <Prefix>TaxDocs</Prefix>
        <Tag>
          <Key>key1</Key>
          <Value>value1</Value>
        </Tag>
        <Tag>
          <Key>key1</Key>
          <Value>value1</Value>
        </Tag>
      </And>
    </Filter>
    <Destination>
      <Bucket>arn:aws:s3:::examptargetbucket</Bucket>
      <Metrics>
        <Status>Enabled</Status>
        <EventThreshold>
          <Minutes>15</Minutes>
        </EventThreshold>
      </Metrics>
      <ReplicationTime>
        <Status>Enabled</Status>
        <Time>
          <Minutes>15</Minutes>
        </Time>
      </ReplicationTime>
    </Destination>
  </Rule>
</ReplicationConfiguration>

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3
PutBucketRequestPayment

Service: Amazon Simple Storage Service

Sets the request payment configuration for a bucket. By default, the bucket owner pays for downloads from the bucket. This configuration parameter enables the bucket owner (only) to specify that the person requesting the download will be charged for the download. For more information, see Requester Pays Buckets.

The following operations are related to PutBucketRequestPayment:

- CreateBucket
- GetBucketRequestPayment

Request Syntax

```xml
PUT /?requestPayment HTTP/1.1
Host: Bucket.s3.amazonaws.com
Content-MD5: ContentMD5
x-amz-expected-bucket-owner: ExpectedBucketOwner
<?xml version="1.0" encoding="UTF-8"?>
<RequestPaymentConfiguration xmlns="http://s3.amazonaws.com/doc/2006-03-01/">
  <Payer><string/></Payer>
</RequestPaymentConfiguration>
```

URI Request Parameters

The request uses the following URI parameters.

**Bucket (p. 326)**

The bucket name.

Required: Yes

**Content-MD5 (p. 326)**

*This header has been deprecated.*

> The base64-encoded 128-bit MD5 digest of the data. You must use this header as a message integrity check to verify that the request body was not corrupted in transit. For more information, see RFC 1864.

**x-amz-expected-bucket-owner (p. 326)**

The account id of the expected bucket owner. If the bucket is owned by a different account, the request will fail with an HTTP 403 (Access Denied) error.

Request Body

The request accepts the following data in XML format.

**RequestPaymentConfiguration (p. 326)**

Root level tag for the RequestPaymentConfiguration parameters.

Required: Yes

**Payer (p. 326)**

Specifies who pays for the download and request fees.
Type: String

Valid Values: Requester | BucketOwner

Required: Yes

Response Syntax

```
HTTP/1.1 200
```

Response Elements

If the action is successful, the service sends back an HTTP 200 response with an empty HTTP body.

Examples

Sample Request

This request creates a Requester Pays bucket named colorpictures.

```
PUT /colorpictures?requestPayment HTTP/1.1
Host: colorpictures.s3.<Region>.amazonaws.com
Content-Length: 173
Date: Wed, 01 Mar 2006 12:00:00 GMT
Authorization: authorization string

<RequestPaymentConfiguration xmlns="http://s3.amazonaws.com/doc/2006-03-01/">
  <Payer>Requester</Payer>
</RequestPaymentConfiguration>
```

Sample Response

Delete the metric configuration with a specified ID, which disables the CloudWatch metrics with the ExampleMetrics value for the FilterId dimension.

```
HTTP/1.1 200 OK
x-amz-id-2: YgIPIfBiKa2bj0KMg95r/0zo3emzU4dzsD4rcKCHQUAdQkf3ShJTOOpXUueF6QKo
x-amz-request-id: 236A8905248E5A01
Date: Wed, 01 Mar 2006 12:00:00 GMT
Location: /colorpictures
Connection: close
Server: AmazonS3
```

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
• AWS SDK for Java
• AWS SDK for JavaScript
• AWS SDK for PHP V3
• AWS SDK for Python
• AWS SDK for Ruby V3
PutBucketTagging
Service: Amazon Simple Storage Service

Sets the tags for a bucket.

Use tags to organize your AWS bill to reflect your own cost structure. To do this, sign up to get your AWS account bill with tag key values included. Then, to see the cost of combined resources, organize your billing information according to resources with the same tag key values. For example, you can tag several resources with a specific application name, and then organize your billing information to see the total cost of that application across several services. For more information, see Cost Allocation and Tagging.

Note
Within a bucket, if you add a tag that has the same key as an existing tag, the new value overwrites the old value. For more information, see Using Cost Allocation in Amazon S3 Bucket Tags.

To use this operation, you must have permissions to perform the s3:PutBucketTagging action. The bucket owner has this permission by default and can grant this permission to others. For more information about permissions, see Permissions Related to Bucket Subresource Operations and Managing Access Permissions to Your Amazon S3 Resources.

PutBucketTagging has the following special errors:

- Error code: InvalidTagError
  Description: The tag provided was not a valid tag. This error can occur if the tag did not pass input validation. For information about tag restrictions, see User-Defined Tag Restrictions and AWS-Generated Cost Allocation Tag Restrictions.

- Error code: MalformedXMLError
  Description: The XML provided does not match the schema.

- Error code: OperationAbortedError
  Description: A conflicting conditional operation is currently in progress against this resource. Please try again.

- Error code: InternalError
  Description: The service was unable to apply the provided tag to the bucket.

The following operations are related to PutBucketTagging:

- GetBucketTagging
- DeleteBucketTagging

Request Syntax

```
PUT /?tagging HTTP/1.1
Host: Bucket.s3.amazonaws.com
Content-MD5: ContentMD5
x-amz-expected-bucket-owner: ExpectedBucketOwner
<?xml version="1.0" encoding="UTF-8"?>
<Tagging xmlns="http://s3.amazonaws.com/doc/2006-03-01/">
  <TagSet>
    <Tag>
      <Key>string</Key>
      <Value>string</Value>
    </Tag>
  </TagSet>
</Tagging>
```
URI Request Parameters

The request uses the following URI parameters.

**Bucket (p. 329)**

The bucket name.

**Required: Yes**

**Content-MD5 (p. 329)**

*This header has been deprecated.*

The base64-encoded 128-bit MD5 digest of the data. You must use this header as a message integrity check to verify that the request body was not corrupted in transit. For more information, see [RFC 1864](https://tools.ietf.org/html/rfc1864).

**x-amz-expected-bucket-owner (p. 329)**

The account id of the expected bucket owner. If the bucket is owned by a different account, the request will fail with an HTTP 403 (Access Denied) error.

Request Body

The request accepts the following data in XML format.

**Tagging (p. 329)**

Root level tag for the Tagging parameters.

**Required: Yes**

**TagSet (p. 329)**

A collection for a set of tags

**Type: Array of Tag (p. 666) data types**

**Required: Yes**

Response Syntax

```
HTTP/1.1 200
```

Response Elements

If the action is successful, the service sends back an HTTP 200 response with an empty HTTP body.

Examples

Sample Request: Add tag set to a bucket

The following request adds a tag set to the existing examplebucket bucket.

```
PUT /tagging HTTP/1.1
Host: examplebucket.s3.<Region>.amazonaws.com
Content-Length: 1660
```
Sample Response

This example illustrates one usage of PutBucketTagging.

HTTP/1.1 204 No Content
x-amz-id-2: YgIP1f81Ka2bj0KMgUAdQkf3ShJOOpXUueF6QKo
x-amz-request-id: 236A8905248E5A01
Date: Wed, 01 Oct 2012 12:00:00 GMT

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3
PutBucketVersioning
Service: Amazon Simple Storage Service

Sets the versioning state of an existing bucket. To set the versioning state, you must be the bucket owner.

You can set the versioning state with one of the following values:

Enabled—Enables versioning for the objects in the bucket. All objects added to the bucket receive a unique version ID.

Suspended—Disables versioning for the objects in the bucket. All objects added to the bucket receive the version ID null.

If the versioning state has never been set on a bucket, it has no versioning state; a GetBucketVersioning request does not return a versioning state value.

If the bucket owner enables MFA Delete in the bucket versioning configuration, the bucket owner must include the x-amz-mfa request header and the Status and the MfaDelete request elements in a request to set the versioning state of the bucket.

Important
If you have an object expiration lifecycle policy in your non-versioned bucket and you want to maintain the same permanent delete behavior when you enable versioning, you must add a noncurrent expiration policy. The noncurrent expiration lifecycle policy will manage the deletes of the noncurrent object versions in the version-enabled bucket. (A version-enabled bucket maintains one current and zero or more noncurrent object versions.) For more information, see Lifecycle and Versioning.

Related Resources
• CreateBucket
• DeleteBucket
• GetBucketVersioning

Request Syntax

```
PUT /?versioning HTTP/1.1
Host: Bucket.s3.amazonaws.com
Content-MD5: ContentMD5
x-amz-mfa: MFA
x-amz-expected-bucket-owner: ExpectedBucketOwner
<?xml version="1.0" encoding="UTF-8"?>
  <MfaDelete>string</MfaDelete>
  <Status>string</Status>
</AddressingConfiguration>
```

URI Request Parameters

The request uses the following URI parameters.

Bucket (p. 332)

The bucket name.

Required: Yes
**Content-MD5 (p. 332)**

*This header has been deprecated.*

> The base64-encoded 128-bit MD5 digest of the data. You must use this header as a message integrity check to verify that the request body was not corrupted in transit. For more information, see RFC 1864.

**x-amz-expected-bucket-owner (p. 332)**

The account id of the expected bucket owner. If the bucket is owned by a different account, the request will fail with an HTTP 403 (Access Denied) error.

**x-amz-mfa (p. 332)**

The concatenation of the authentication device's serial number, a space, and the value that is displayed on your authentication device.

**Request Body**

The request accepts the following data in XML format.

**VersioningConfiguration (p. 332)**

Root level tag for the VersioningConfiguration parameters.

Required: Yes

**MFADelete (p. 332)**

Specifies whether MFA delete is enabled in the bucket versioning configuration. This element is only returned if the bucket has been configured with MFA delete. If the bucket has never been so configured, this element is not returned.

Type: String

Valid Values: Enabled | Disabled

Required: No

**Status (p. 332)**

The versioning state of the bucket.

Type: String

Valid Values: Enabled | Suspended

Required: No

**Response Syntax**

```
HTTP/1.1 200
```

**Response Elements**

If the action is successful, the service sends back an HTTP 200 response with an empty HTTP body.

**Examples**

**Sample Request**

The following request enables versioning for the specified bucket.
**Sample Request**

The following request suspends versioning for the specified bucket.

```plaintext
PUT /?versioning HTTP/1.1
Host: bucket.s3.<Region>.amazonaws.com
Date: Wed, 12 Oct 2009 17:50:00 GMT
Authorization: authorization string
Content-Type: text/plain
Content-Length: 124

  <Status>Suspended</Status>
</VersioningConfiguration>
```

**Sample Response**

This example illustrates one usage of PutBucketVersioning.

```
HTTP/1.1 200 OK
x-amz-id-2: YgIPIfBiKa2bj0KMg95r/0zo3emzU4dzsD4rcKCHQUAdQkf3ShJTOOpXUueF6QKo
x-amz-request-id: 236A8905248E5A01
Date: Wed, 01 Mar 2006 12:00:00 GMT
```

**Sample Request**

The following request enables versioning and MFA Delete on a bucket. Note the space between [SerialNumber] and [TokenCode] and that you must include Status whenever you use MfaDelete.

```plaintext
PUT /?versioning HTTP/1.1
Host: bucket.s3.<Region>.amazonaws.com
Date: Wed, 01 Mar 2006 12:00:00 GMT
Authorization: authorization string
Content-Type: text/plain
Content-Length: 124

  <Status>Enabled</Status>
</VersioningConfiguration>
```

**Sample Response**

This example illustrates one usage of PutBucketVersioning.

```
HTTP/1.1 200 OK
x-amz-id-2: YgIPIfBiKa2bj0KMg95r/0zo3emzU4dzsD4rcKCHQUAdQkf3ShJTOOpXUueF6QKo
x-amz-request-id: 236A8905248E5A01
Date: Wed, 01 Mar 2006 12:00:00 GMT
```
    <Status>Enabled</Status>
    <MfaDelete>Enabled</MfaDelete>
</VersioningConfiguration>

Sample Response

This example illustrates one usage of PutBucketVersioning.

HTTPS/1.1 200 OK
x-amz-id-2: YgIPIfBiKa2bj0KMg95r/0zo3emzU4dzsD4rcRCHQUAdQkf3ShJTOopXUueF6QRo
x-amz-request-id: 236A8905248E5A01
Date: Wed, 01 Mar 2006 12:00:00 GMT

Location: /colorpictures
Content-Length: 0
Connection: close
Server: AmazonS3

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3
PutBucketWebsite
Service: Amazon Simple Storage Service

Sets the configuration of the website that is specified in the website subresource. To configure a bucket as a website, you can add this subresource on the bucket with website configuration information such as the file name of the index document and any redirect rules. For more information, see Hosting Websites on Amazon S3.

This PUT operation requires the S3:PutBucketWebsite permission. By default, only the bucket owner can configure the website attached to a bucket; however, bucket owners can allow other users to set the website configuration by writing a bucket policy that grants them the S3:PutBucketWebsite permission.

To redirect all website requests sent to the bucket's website endpoint, you add a website configuration with the following elements. Because all requests are sent to another website, you don't need to provide index document name for the bucket.

• WebsiteConfiguration
• RedirectAllRequestsTo
• HostName
• Protocol

If you want granular control over redirects, you can use the following elements to add routing rules that describe conditions for redirecting requests and information about the redirect destination. In this case, the website configuration must provide an index document for the bucket, because some requests might not be redirected.

• WebsiteConfiguration
• IndexDocument
• Suffix
• ErrorDocument
• Key
• RoutingRules
• RoutingRule
• Condition
• HttpStatusCodeReturnedEquals
• KeyPrefixEquals
• Redirect
• Protocol
• HostName
• ReplaceKeyPrefixWith
• ReplaceKeyWith
• HttpRedirectCode

Amazon S3 has a limitation of 50 routing rules per website configuration. If you require more than 50 routing rules, you can use object redirect. For more information, see Configuring an Object Redirect in the Amazon Simple Storage Service Developer Guide.

Request Syntax

PUT /?website HTTP/1.1
Host: Bucket.s3.amazonaws.com
Content-MD5: ContentMD5
x-amz-expected-bucket-owner: ExpectedBucketOwner

<?xml version="1.0" encoding="UTF-8"?>
<WebsiteConfiguration xmlns="http://s3.amazonaws.com/doc/2006-03-01/"
  <ErrorDocument>
    <Key>string</Key>
  </ErrorDocument>
  <IndexDocument>
    <Suffix>string</Suffix>
  </IndexDocument>
  <RedirectAllRequestsTo>
    <HostName>string</HostName>
    <Protocol>string</Protocol>
  </RedirectAllRequestsTo>
  <RoutingRules>
    <RoutingRule>
      <Condition>
        <HttpErrorCodeReturnedEquals>string</HttpErrorCodeReturnedEquals>
        <KeyPrefixEquals>string</KeyPrefixEquals>
      </Condition>
      <Redirect>
        <HostName>string</HostName>
        <HttpRedirectCode>string</HttpRedirectCode>
        <Protocol>string</Protocol>
        <ReplaceKeyPrefixWith>string</ReplaceKeyPrefixWith>
        <ReplaceKeyWith>string</ReplaceKeyWith>
      </Redirect>
    </RoutingRule>
  </RoutingRules>
</WebsiteConfiguration>

URI Request Parameters

The request uses the following URI parameters.

Bucket (p. 336)

The bucket name.

Required: Yes

Content-MD5 (p. 336)

This header has been deprecated.

The base64-encoded 128-bit MD5 digest of the data. You must use this header as a message integrity check to verify that the request body was not corrupted in transit. For more information, see RFC 1864.

x-amz-expected-bucket-owner (p. 336)

The account id of the expected bucket owner. If the bucket is owned by a different account, the request will fail with an HTTP 403 (Access Denied) error.

Request Body

The request accepts the following data in XML format.

WebsiteConfiguration (p. 336)

Root level tag for the WebsiteConfiguration parameters.
Required: Yes

**ErrorDocument (p. 336)**

The name of the error document for the website.

Type: ErrorDocument (p. 562) data type

Required: No

**IndexDocument (p. 336)**

The name of the index document for the website.

Type: IndexDocument (p. 569) data type

Required: No

**RedirectAllRequestsTo (p. 336)**

The redirect behavior for every request to this bucket's website endpoint.

**Important**

If you specify this property, you can't specify any other property.

Type: RedirectAllRequestsTo (p. 634) data type

Required: No

**RoutingRules (p. 336)**

Rules that define when a redirect is applied and the redirect behavior.

Type: Array of RoutingRule (p. 646) data types

Required: No

### Response Syntax

```
HTTP/1.1 200
```

### Response Elements

If the action is successful, the service sends back an HTTP 200 response with an empty HTTP body.

### Examples

**Example 1: Configure bucket as a website (add website configuration)**

The following request configures a bucket example.com as a website. The configuration in the request specifies index.html as the index document. It also specifies the optional error document, SomeErrorDocument.html.

```
PUT ?website HTTP/1.1
Host: example.com.s3.<Region>.amazonaws.com
Content-Length: 256
Date: Thu, 27 Jan 2011 12:00:00 GMT
Authorization: signatureValue

<WebsiteConfiguration xmlns='http://s3.amazonaws.com/doc/2006-03-01/'>
  <IndexDocument>
    <Suffix>index.html</Suffix>
  </IndexDocument>
</WebsiteConfiguration>
```

API Version 2006-03-01

338
Sample Response

This example illustrates one usage of PutBucketWebsite.

HTTP/1.1 200 OK
x-amz-id-2: YgIPIfBiKa2bj0KMgUAdQkf3ShJTOOpUXUueF6QKo
x-amz-request-id: 80CD4368BD211111
Date: Thu, 27 Jan 2011 00:00:00 GMT
Content-Length: 0
Server: AmazonS3

Example 2: Configure bucket as a website but redirect all requests

The following request configures a bucket www.example.com as a website. However, the configuration specifies that all GET requests for the www.example.com bucket's website endpoint will be redirected to host example.com. This redirect can be useful when you want to serve requests for both http://www.example.com and http://example.com, but you want to maintain the website content in only one bucket, in this case, example.com.

PUT ?website HTTP/1.1
Host: www.example.com.s3.<Region>.amazonaws.com
Content-Length: length-value
Date: Thu, 27 Jan 2011 12:00:00 GMT
Authorization: signatureValue

<WebsiteConfiguration xmlns='http://s3.amazonaws.com/doc/2006-03-01/'>
  <RedirectAllRequestsTo>
    <HostName>example.com</HostName>
  </RedirectAllRequestsTo>
</WebsiteConfiguration>

Example 3: Configure bucket as a website and specify optional redirection rules

Example 1 is the simplest website configuration. It configures a bucket as a website by providing only an index document and an error document. You can further customize the website configuration by adding routing rules that redirect requests for one or more objects. For example, suppose that your bucket contained the following objects:

- index.html
- docs/article1.html
- docs/article2.html

If you decided to rename the folder from docs/ to documents/, you would need to redirect requests for prefix /docs to documents/. For example, a request for docs/article1.html will need to be redirected to documents/article1.html.

In this case, you update the website configuration and add a routing rule as shown in the following request.
Example 4: Configure a bucket as a website and redirect errors

You can use a routing rule to specify a condition that checks for a specific HTTP error code. When a page request results in this error, you can optionally reroute requests. For example, you might route requests to another host and optionally process the error. The routing rule in the following requests redirects requests to an EC2 instance in the event of an HTTP error 404. For illustration, the redirect also inserts an object key prefix report-404/ in the redirect. For example, if you request a page ExamplePage.html and it results in an HTTP 404 error, the request is routed to a page report-404/testPage.html on the specified EC2 instance. If there is no routing rule and the HTTP error 404 occurred, then Error.html would be returned.

```xml
<WebsiteConfiguration xmlns='http://s3.amazonaws.com/doc/2006-03-01/'>
  <IndexDocument>
    <Suffix>index.html</Suffix>
  </IndexDocument>
  <ErrorDocument>
    <Key>Error.html</Key>
  </ErrorDocument>
  <RoutingRules>
    <RoutingRule>
      <Condition>
        <HttpErrorCodeReturnedEquals>404</HttpErrorCodeReturnedEquals>
      </Condition>
      <Redirect>
        <HostName>ec2-11-22-333-44.compute-1.amazonaws.com</HostName>
        <ReplaceKeyPrefixWith>report-404/</ReplaceKeyPrefixWith>
      </Redirect>
    </RoutingRule>
  </RoutingRules>
</WebsiteConfiguration>
```
Suppose you have the following pages in your bucket:

- images/photo1.jpg
- images/photo2.jpg
- images/photo3.jpg

Now you want to route requests for all pages with the images/ prefix to go to a single page, errorpage.html. You can add a website configuration to your bucket with the routing rule shown in the following request.

```
PUT ?website HTTP/1.1
Host: www.example.com.s3.<Region>.amazonaws.com
Content-Length: 481
Date: Thu, 27 Jan 2011 12:00:00 GMT
Authorization: signatureValue

<WebsiteConfiguration xmlns='http://s3.amazonaws.com/doc/2006-03-01/'>
  <IndexDocument>
    <Suffix>index.html</Suffix>
  </IndexDocument>
  <ErrorDocument>
    <Key>Error.html</Key>
  </ErrorDocument>
  <RoutingRules>
    <RoutingRule>
      <Condition>
        <KeyPrefixEquals>images/</KeyPrefixEquals>
      </Condition>
      <Redirect>
        <ReplaceKeyWith>errorpage.html</ReplaceKeyWith>
      </Redirect>
    </RoutingRule>
  </RoutingRules>
</WebsiteConfiguration>
```

**See Also**

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3
PutObject
Service: Amazon Simple Storage Service

Adds an object to a bucket. You must have WRITE permissions on a bucket to add an object to it.

Amazon S3 never adds partial objects; if you receive a success response, Amazon S3 added the entire object to the bucket.

Amazon S3 is a distributed system. If it receives multiple write requests for the same object simultaneously, it overwrites all but the last object written. Amazon S3 does not provide object locking; if you need this, make sure to build it into your application layer or use versioning instead.

To ensure that data is not corrupted traversing the network, use the Content-MD5 header. When you use this header, Amazon S3 checks the object against the provided MD5 value and, if they do not match, returns an error. Additionally, you can calculate the MD5 while putting an object to Amazon S3 and compare the returned ETag to the calculated MD5 value.

**Note**
The Content-MD5 header is required for any request to upload an object with a retention period configured using Amazon S3 Object Lock. For more information about Amazon S3 Object Lock, see Amazon S3 Object Lock Overview in the Amazon Simple Storage Service Developer Guide.

Server-side Encryption

You can optionally request server-side encryption. With server-side encryption, Amazon S3 encrypts your data as it writes it to disks in its data centers and decrypts the data when you access it. You have the option to provide your own encryption key or use AWS managed encryption keys. For more information, see Using Server-Side Encryption.

Access Control List (ACL)-Specific Request Headers

You can use headers to grant ACL-based permissions. By default, all objects are private. Only the owner has full access control. When adding a new object, you can grant permissions to individual AWS accounts or to predefined groups defined by Amazon S3. These permissions are then added to the ACL on the object. For more information, see Access Control List (ACL) Overview and Managing ACLs Using the REST API.

Storage Class Options

By default, Amazon S3 uses the STANDARD Storage Class to store newly created objects. The STANDARD storage class provides high durability and high availability. Depending on performance needs, you can specify a different Storage Class. Amazon S3 on Outposts only uses the OUTPOSTS Storage Class. For more information, see Storage Classes in the Amazon S3 Service Developer Guide.

Versioning

If you enable versioning for a bucket, Amazon S3 automatically generates a unique version ID for the object being stored. Amazon S3 returns this ID in the response. When you enable versioning for a bucket, if Amazon S3 receives multiple write requests for the same object simultaneously, it stores all of the objects.

For more information about versioning, see Adding Objects to Versioning Enabled Buckets. For information about returning the versioning state of a bucket, see GetBucketVersioning.

Related Resources

- CopyObject
- DeleteObject
Request Syntax

```
PUT /Key+ HTTP/1.1
Host: Bucket.s3.amazonaws.com
x-amz-acl: ACL
Cache-Control: CacheControl
Content-Disposition: ContentDisposition
Content-Encoding: ContentEncoding
Content-Language: ContentLanguage
Content-Length: ContentLength
Content-MD5: ContentMD5
Content-Type: ContentType
Expires: Expires
x-amz-grant-full-control: GrantFullControl
x-amz-grant-read: GrantRead
x-amz-grant-read-acp: GrantReadACP
x-amz-grant-write-acp: GrantWriteACP
x-amz-storage-class: StorageClass
x-amz-website-redirect-location: WebsiteRedirectLocation
x-amz-server-side-encryption: ServerSideEncryption
x-amz-server-side-encryption-customer-algorithm: SSECustomerAlgorithm
x-amz-server-side-encryption-customer-key: SSECustomerKey
x-amz-server-side-encryption-aws-kms-key-id: SSESKEncryptionContext
x-amz-server-side-encryption-context: SSEKMSKey
x-amz-request-payer: RequestPayer
x-amz-tagging: Tagging
x-amz-object-lock-mode: ObjectLockMode
x-amz-object-lock-retain-until-date: ObjectLockRetainUntilDate
x-amz-object-lock-legal-hold: ObjectLockLegalHoldStatus
x-amz-expected-bucket-owner: ExpectedBucketOwner
```

Body

URI Request Parameters

The request uses the following URI parameters.

**Bucket (p. 344)**

The bucket name to which the PUT operation was initiated.

When using this API with an access point, you must direct requests to the access point hostname. The access point hostname takes the form **AccessPointName-AccountId.s3-accesspoint.Region.amazonaws.com**. When using this operation with an access point through the AWS SDKs, you provide the access point ARN in place of the bucket name. For more information about access point ARNs, see Using Access Points in the Amazon Simple Storage Service Developer Guide.

When using this API with Amazon S3 on Outposts, you must direct requests to the S3 on Outposts hostname. The S3 on Outposts hostname takes the form **AccessPointName-AccountId.outpostId.s3-outposts.Region.amazonaws.com**. When using this operation using S3 on Outposts through the AWS SDKs, you provide the Outposts bucket ARN in place of the bucket name. For more information about S3 on Outposts ARNs, see Using S3 on Outposts in the Amazon Simple Storage Service Developer Guide.

Required: Yes

**Cache-Control (p. 344)**

Can be used to specify caching behavior along the request/reply chain. For more information, see http://www.w3.org/Protocols/rfc2616/rfc2616-sec14.html#sec14.9.
Content-Disposition (p. 344)

Specifies presentational information for the object. For more information, see http://www.w3.org/Protocols/rfc2616/rfc2616-sec19.html#sec19.5.1.

Content-Encoding (p. 344)

Specifies what content encodings have been applied to the object and thus what decoding mechanisms must be applied to obtain the media-type referenced by the Content-Type header field. For more information, see http://www.w3.org/Protocols/rfc2616/rfc2616-sec14.html#sec14.11.

Content-Language (p. 344)

The language the content is in.

Content-Length (p. 344)

Size of the body in bytes. This parameter is useful when the size of the body cannot be determined automatically. For more information, see http://www.w3.org/Protocols/rfc2616/rfc2616-sec14.html#sec14.13.

Content-MD5 (p. 344)

The base64-encoded 128-bit MD5 digest of the message (without the headers) according to RFC 1864. This header can be used as a message integrity check to verify that the data is the same data that was originally sent. Although it is optional, we recommend using the Content-MD5 mechanism as an end-to-end integrity check. For more information about REST request authentication, see REST Authentication.

Content-Type (p. 344)

A standard MIME type describing the format of the contents. For more information, see http://www.w3.org/Protocols/rfc2616/rfc2616-sec14.html#sec14.17.

Expires (p. 344)

The date and time at which the object is no longer cacheable. For more information, see http://www.w3.org/Protocols/rfc2616/rfc2616-sec14.html#sec14.21.

Key (p. 344)

Object key for which the PUT operation was initiated.

Length Constraints: Minimum length of 1.

Required: Yes

x-amz-acl (p. 344)

The canned ACL to apply to the object. For more information, see Canned ACL.

This action is not supported by Amazon S3 on Outposts.

Valid Values: private | public-read | public-read-write | authenticated-read | aws-exec-read | bucket-owner-read | bucket-owner-full-control

x-amz-expected-bucket-owner (p. 344)

The account id of the expected bucket owner. If the bucket is owned by a different account, the request will fail with an HTTP 403 (Access Denied) error.

x-amz-grant-full-control (p. 344)

Gives the grantee READ, READ_ACP, and WRITE_ACP permissions on the object.

This action is not supported by Amazon S3 on Outposts.

x-amz-grant-read (p. 344)

Allows grantee to read the object data and its metadata.
This action is not supported by Amazon S3 on Outposts.

**x-amz-grant-read-acp (p. 344)**

Allows grantee to read the object ACL.

This action is not supported by Amazon S3 on Outposts.

**x-amz-grant-write-acp (p. 344)**

Allows grantee to write the ACL for the applicable object.

This action is not supported by Amazon S3 on Outposts.

**x-amz-object-lock-legal-hold (p. 344)**

Specifies whether a legal hold will be applied to this object. For more information about S3 Object Lock, see [Object Lock](#).

Valid Values: ON | OFF

**x-amz-object-lock-mode (p. 344)**

The Object Lock mode that you want to apply to this object.

Valid Values: GOVERNANCE | COMPLIANCE

**x-amz-object-lock-retain-until-date (p. 344)**

The date and time when you want this object's Object Lock to expire.

**x-amz-request-payer (p. 344)**

Confirms that the requester knows that they will be charged for the request. Bucket owners need not specify this parameter in their requests. For information about downloading objects from requester pays buckets, see [Downloading Objects in Requestor Pays Buckets](#) in the Amazon S3 Developer Guide.

Valid Values: requester

**x-amz-server-side-encryption (p. 344)**

The server-side encryption algorithm used when storing this object in Amazon S3 (for example, AES256, aws:kms).

Valid Values: AES256 | aws:kms

**x-amz-server-side-encryption-aws-kms-key-id (p. 344)**

If `x-amz-server-side-encryption` is present and has the value of `aws:kms`, this header specifies the ID of the AWS Key Management Service (AWS KMS) symmetrical customer managed customer master key (CMK) that was used for the object.

If the value of `x-amz-server-side-encryption` is `aws:kms`, this header specifies the ID of the symmetric customer managed AWS KMS CMK that will be used for the object. If you specify `x-amz-server-side-encryption:aws:kms`, but do not provide `x-amz-server-side-encryption-aws-kms-key-id`, Amazon S3 uses the AWS managed CMK in AWS to protect the data.

**x-amz-server-side-encryption-context (p. 344)**

Specifies the AWS KMS Encryption Context to use for object encryption. The value of this header is a base64-encoded UTF-8 string holding JSON with the encryption context key-value pairs.

**x-amz-server-side-encryption-customer-algorithm (p. 344)**

Specifies the algorithm to use to when encrypting the object (for example, AES256).
x-amz-server-side-encryption-customer-key (p. 344)

Specifies the customer-provided encryption key for Amazon S3 to use in encrypting data. This value is used to store the object and then it is discarded; Amazon S3 does not store the encryption key. The key must be appropriate for use with the algorithm specified in the x-amz-server-side-encryption-customer-algorithm header.

x-amz-server-side-encryption-customer-key-MD5 (p. 344)

Specifies the 128-bit MD5 digest of the encryption key according to RFC 1321. Amazon S3 uses this header for a message integrity check to ensure that the encryption key was transmitted without error.

x-amz-storage-class (p. 344)

By default, Amazon S3 uses the STANDARD Storage Class to store newly created objects. The STANDARD storage class provides high durability and high availability. Depending on performance needs, you can specify a different Storage Class. Amazon S3 on Outposts only uses the OUTPOSTS Storage Class. For more information, see Storage Classes in the Amazon S3 Service Developer Guide.

Valid Values: STANDARD | REDUCED_REDUNDANCY | STANDARD_IA | ONEZONE_IA | INTELLIGENT_TIERING | GLACIER | DEEP_ARCHIVE | OUTPOSTS

x-amz-tagging (p. 344)

The tag-set for the object. The tag-set must be encoded as URL Query parameters. (For example, "Key1=Value1")

x-amz-website-redirect-location (p. 344)

If the bucket is configured as a website, redirects requests for this object to another object in the same bucket or to an external URL. Amazon S3 stores the value of this header in the object metadata. For information about object metadata, see Object Key and Metadata.

In the following example, the request header sets the redirect to an object (anotherPage.html) in the same bucket:

x-amz-website-redirect-location: /anotherPage.html

In the following example, the request header sets the object redirect to another website:

x-amz-website-redirect-location: http://www.example.com/

For more information about website hosting in Amazon S3, see Hosting Websites on Amazon S3 and How to Configure Website Page Redirects.

Request Body

The request accepts the following binary data.

Body (p. 344)

Response Syntax

```
HTTP/1.1 200
x-amz-expiration: Expiration
ETag: ETag
x-amz-server-side-encryption: ServerSideEncryption
x-amz-version-id: VersionId
x-amz-server-side-encryption-customer-algorithm: SSECustomerAlgorithm
x-amz-server-side-encryption-customer-key-MD5: SSECustomerKeyMD5
```
Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The response returns the following HTTP headers.

ETag (p. 347)
Entity tag for the uploaded object.

x-amz-expiration (p. 347)
If the expiration is configured for the object (see PutBucketLifecycleConfiguration), the response includes this header. It includes the expiry-date and rule-id key-value pairs that provide information about object expiration. The value of the rule-id is URL encoded.

x-amz-request-charged (p. 347)
If present, indicates that the requester was successfully charged for the request.

Valid Values: requester

x-amz-server-side-encryption (p. 347)
If you specified server-side encryption either with an AWS KMS customer master key (CMK) or Amazon S3-managed encryption key in your PUT request, the response includes this header. It confirms the encryption algorithm that Amazon S3 used to encrypt the object.

Valid Values: AES256 | aws:kms

x-amz-server-side-encryption-aws-kms-key-id (p. 347)
If x-amz-server-side-encryption is present and has the value of aws:kms, this header specifies the ID of the AWS Key Management Service (AWS KMS) symmetric customer managed customer master key (CMK) that was used for the object.

x-amz-server-side-encryption-context (p. 347)
If present, specifies the AWS KMS Encryption Context to use for object encryption. The value of this header is a base64-encoded UTF-8 string holding JSON with the encryption context key-value pairs.

x-amz-server-side-encryption-customer-algorithm (p. 347)
If server-side encryption with a customer-provided encryption key was requested, the response will include this header confirming the encryption algorithm used.

x-amz-server-side-encryption-customer-key-MD5 (p. 347)
If server-side encryption with a customer-provided encryption key was requested, the response will include this header to provide round-trip message integrity verification of the customer-provided encryption key.

x-amz-version-id (p. 347)
Version of the object.

Examples

Example 1: Upload an object

The following request stores the my-image.jpg file in the myBucket bucket.
PUT /my-image.jpg HTTP/1.1
Host: myBucket.s3.<Region>.amazonaws.com
Date: Wed, 12 Oct 2009 17:50:00 GMT
Authorization: authorization string
Content-Type: text/plain
Content-Length: 11434
x-amz-meta-author: Janet
Expect: 100-continue

[11434 bytes of object data]

Sample Response: Versioning suspended

This example illustrates one usage of PutObject.

HTTP/1.1 100 Continue
HTTP/1.1 200 OK
x-amz-id-2: LriYPLdmOdaIfgSm/F1YsVit1LW94/xUQxMsF7x1Eb1a0wiI01xl+zbwZ163pt7
x-amz-request-id: 0A49CE4060975EAC
Date: Wed, 12 Oct 2009 17:50:00 GMT
ETag: "1b2cf535f27731c974343645a3985328"
Content-Length: 0
Connection: close
Server: AmazonS3

Sample Response: Expiration rule createdUsingLifecyle configuration

If an expiration rule that was created on the bucket using lifecycle configuration applies to the object, you get a response with an x-amz-expiration header, as shown in the following response. For more information, see Transitioning Objects: General Considerations.

HTTP/1.1 100 Continue
HTTP/1.1 200 OK
x-amz-id-2: LriYPLdmOdaIfgSm/F1YsVit1LW94/xUQxMsF7x1Eb1a0wiI01xl+zbwZ163pt7
x-amz-request-id: 0A49CE4060975EAC
Date: Wed, 12 Oct 2009 17:50:00 GMT
x-amz-expiration: expiry-date="Fri, 23 Dec 2012 00:00:00 GMT", rule-id="1"
ETag: "1b2cf535f27731c974343645a3985328"
Content-Length: 0
Connection: close
Server: AmazonS3

Sample Response: Versioning enabled

If the bucket has versioning enabled, the response includes the x-amz-version-id header.

HTTP/1.1 100 Continue
HTTP/1.1 200 OK
x-amz-id-2: LriYPLdmOdaIfgSm/F1YsVit1LW94/xUQxMsF7x1Eb1a0wiI01xl+zbwZ163pt7
x-amz-request-id: 0A49CE4060975EAC
x-amz-version-id: 43jfkod08493jnFJD9fjJj3HHNVfsqUHFDNsid038jfdsjGFDSIRp
Example 2: Specifying the Reduced Redundancy Storage Class

The following request stores the image, my-image.jpg, in the myBucket bucket. The request specifies the x-amz-storage-class header to request that the object is stored using the REDUCED_REDUNDANCY storage class.

PUT /my-image.jpg HTTP/1.1
Host: myBucket.s3.<Region>.amazonaws.com
Date: Wed, 12 Oct 2009 17:50:00 GMT
Authorization: authorization string
Content-Type: image/jpeg
Content-Length: 11434
Expect: 100-continue
x-amz-storage-class: REDUCED_REDUNDANCY

Sample Response

This example illustrates one usage of PutObject.

HTTP/1.1 100 Continue
HTTP/1.1 200 OK
x-amz-id-2: LriYPLdmOdAilfgSm/F1YsVit1LW94/xUQxMsF7x1E1a0WiIOIxl+zbwZ163pt7
x-amz-request-id: 0A49CE4060975EAC
Date: Wed, 12 Oct 2009 17:50:00 GMT
ETag: "1b2cf535f27731c9771645a39863328"
Content-Length: 0
Connection: close
Server: AmazonS3

Example 3: Uploading an object and specifying access permissions explicitly

The following request stores the TestObject.txt file in the myBucket bucket. The request specifies various ACL headers to grant permission to AWS accounts that are specified with a canonical user ID and an email address.

PUT TestObject.txt HTTP/1.1
Host: myBucket.s3.<Region>.amazonaws.com
x-amz-date: Fri, 13 Apr 2012 05:40:14 GMT
Authorization: authorization string
x-amz-grant-write-acp: id=8a6925ce4adf588a4532142d3f74dd8c71fa124ExampleCanonicalUserID
x-amz-grant-full-control: emailAddress="ExampleUser@amazon.com"
x-amz-grant-write: emailAddress="ExampleUser1@amazon.com",
emailAddress="ExampleUser2@amazon.com"
Content-Length: 300
Expect: 100-continue
Connection: Keep-Alive


**Sample Response**

This example illustrates one usage of PutObject.

```
HTTP/1.1 200 OK
x-amz-id-2: RUxG2sZJUfS+ezeAS2i0Xj6w/ST6xqF/8pFHNjTjTrECW56SCAUWGg+7QLVoj1GH
x-amz-request-id: 8D017A90827290BA
Date: Fri, 13 Apr 2012 05:40:25 GMT
ETag: "dd038b344cf9553547f8b395a814b274"
Content-Length: 0
Server: AmazonS3
```

**Example 4: Using a canned ACL to set access permissions**

The following request stores the `TestObject.txt` file in the `myBucket` bucket. The request uses an `x-amz-acl` header to specify a canned ACL that grants READ permission to the public.

```
...Object data in the body...
PUT TestObject.txt HTTP/1.1
Host: myBucket.s3.<Region>.amazonaws.com
x-amz-date: Fri, 13 Apr 2012 05:54:57 GMT
x-amz-acl: public-read
Authorization: authorization string
Content-Length: 300
Expect: 100-continue
Connection: Keep-Alive
...Object data in the body...
```

**Sample Response**

This example illustrates one usage of PutObject.

```
HTTP/1.1 200 OK
x-amz-id-2: Yd6PSJxJFQeTYJ/3dDO7miq3fVMXXW0S2HiJo3WFs4bz6oe2QCVXasxXLZdMfASd
x-amz-request-id: 80DF413BB3D28A25
Date: Fri, 13 Apr 2012 05:54:59 GMT
ETag: "dd038b344cf9553547f8b395a814b274"
Content-Length: 0
Server: AmazonS3
```

**Example 5: Upload an object (Request server-side encryption using a customer-provided encryption key)**

This example of an upload object requests server-side encryption and provides an encryption key.

```
PUT /example-object HTTP/1.1
Host: example-bucket.s3.<Region>.amazonaws.com
Accept: */*
Authorization: authorization string
```

API Version 2006-03-01
Sample Response

In the response, Amazon S3 returns the encryption algorithm and MD5 of the encryption key that you specified when uploading the object. The ETag that is returned is not the MD5 of the object.

```
HTTP/1.1 200 OK
x-amz-id-2: 7qoYGN7uMuFuYS6m7a4lszH6in+hccE+4DXPmdZ7C9KqucjnZC1gI5mshai6fbMG
x-amz-request-id: 06437EDD40C407C7
Date: Wed, 28 May 2014 19:31:12 GMT
x-amz-server-side-encryption-customer-algorithm: AES256
x-amz-server-side-encryption-customer-key-MD5: ZjQrne1X/iTcskbY2example
ETag: "ae89237c20e759c5f479ece02c642f59"
```

Example 6: Upload an object and specify tags

This example of an upload object request specifies the optional `x-amz-tagging` header to add tags to the object.

After the object is created, Amazon S3 stores the specified object tags in the tagging subresource that is associated with the object. For more information about tagging, see Object Tagging and Access Control Policies in the Amazon Simple S3 Developer Guide.

```
PUT /example-object HTTP/1.1
Host: example-bucket.s3.<Region>.amazonaws.com
Accept: */*
Authorization: authorization string
Date: Thu, 22 Sep 2016 21:58:13 GMT
x-amz-tagging: tag1=value1&tag2=value2

[... bytes of object data]
```

Sample Response

This example illustrates one usage of PutObject.

```
HTTP/1.1 200 OK
x-amz-id-2: 7qoYGN7uMuFuYS6m7a4lszH6in+hccE+4DXPmdZ7C9KqucjnZC1gI5mshai6fbMG
x-amz-request-id: 06437EDD40C407C7
Date: Thu, 22 Sep 2016 21:58:17 GMT
```

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3
PutObjectAcl
Service: Amazon Simple Storage Service

Uses the acl subresource to set the access control list (ACL) permissions for a new or existing object in an S3 bucket. You must have WRITE_ACP permission to set the ACL of an object. For more information, see What permissions can I grant? in the Amazon Simple Storage Service Developer Guide.

This action is not supported by Amazon S3 on Outposts.

Depending on your application needs, you can choose to set the ACL on an object using either the request body or the headers. For example, if you have an existing application that updates a bucket ACL using the request body, you can continue to use that approach. For more information, see Access Control List (ACL) Overview in the Amazon S3 Developer Guide.

Access Permissions

You can set access permissions using one of the following methods:

- Specify a canned ACL with the x-amz-acl request header. Amazon S3 supports a set of predefined ACLs, known as canned ACLs. Each canned ACL has a predefined set of grantees and permissions. Specify the canned ACL name as the value of x-amz-acl. If you use this header, you cannot use other access control-specific headers in your request. For more information, see Canned ACL.

- Specify access permissions explicitly with the x-amz-grant-read, x-amz-grant-read-acp, x-amz-grant-write-acp, and x-amz-grant-full-control headers. When using these headers, you specify explicit access permissions and grantees (AWS accounts or Amazon S3 groups) who will receive the permission. If you use these ACL-specific headers, you cannot use x-amz-acl header to set a canned ACL. These parameters map to the set of permissions that Amazon S3 supports in an ACL. For more information, see Access Control List (ACL) Overview.

You specify each grantee as a type=value pair, where the type is one of the following:
- id – if the value specified is the canonical user ID of an AWS account
- uri – if you are granting permissions to a predefined group
- emailAddress – if the value specified is the email address of an AWS account

Note
Using email addresses to specify a grantee is only supported in the following AWS Regions:
- US East (N. Virginia)
- US West (N. California)
- US West (Oregon)
- Asia Pacific (Singapore)
- Asia Pacific (Sydney)
- Asia Pacific (Tokyo)
- Europe (Ireland)
- South America (São Paulo)

For a list of all the Amazon S3 supported Regions and endpoints, see Regions and Endpoints in the AWS General Reference.

For example, the following x-amz-grant-read header grants list objects permission to the two AWS accounts identified by their email addresses.

x-amz-grant-read: emailAddress="xyz@amazon.com",
emailAddress="abc@amazon.com"

You can use either a canned ACL or specify access permissions explicitly. You cannot do both.

Grantee Values
You can specify the person (grantee) to whom you’re assigning access rights (using request elements) in the following ways:

- By the person's ID:

  `<Grantee xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:type="CanonicalUser"><ID><ID></ID><DisplayName><GranteesEmail></DisplayName></Grantee`
  
  DisplayName is optional and ignored in the request.

- By URI:


- By Email address:

  `<Grantee xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:type="AmazonCustomerByEmail"><EmailAddress><Grantees@email.com></EmailAddress></Grantee`

  The grantee is resolved to the CanonicalUser and, in a response to a GET Object acl request, appears as the CanonicalUser.

  **Note**
  Using email addresses to specify a grantee is only supported in the following AWS Regions:
  - US East (N. Virginia)
  - US West (N. California)
  - US West (Oregon)
  - Asia Pacific (Singapore)
  - Asia Pacific (Sydney)
  - Asia Pacific (Tokyo)
  - Europe (Ireland)
  - South America (São Paulo)
  For a list of all the Amazon S3 supported Regions and endpoints, see [Regions and Endpoints](https://aws.amazon.com/regions/) in the AWS General Reference.

**Versioning**

The ACL of an object is set at the object version level. By default, PUT sets the ACL of the current version of an object. To set the ACL of a different version, use the `versionId` subresource.

**Related Resources**

- CopyObject
- GetObject

**Request Syntax**

```
PUT /{Key+}?acl&versionId=VersionId HTTP/1.1
Host: Bucket.s3.amazonaws.com
x-amz-acl: ACL
Content-MD5: ContentMD5
x-amz-grant-full-control: GrantFullControl
x-amz-grant-read: GrantRead
```
x-amz-grant-read-acp: GrantReadACP
x-amz-grant-write: GrantWrite
x-amz-grant-write-acp: GrantWriteACP
x-amz-request-payer: RequestPayer
x-amz-expected-bucket-owner: ExpectedBucketOwner

<?xml version="1.0" encoding="UTF-8"?>
  <AccessControlList>
    <Grant>
      <Grantee>
        <DisplayName>string</DisplayName>
        <EmailAddress>string</EmailAddress>
        <ID>string</ID>
        <xsi:type>string</xsi:type>
        <URI>string</URI>
      </Grantee>
      <Permission>string</Permission>
    </Grant>
  </AccessControlList>
  <Owner>
    <DisplayName>string</DisplayName>
    <ID>string</ID>
  </Owner>
</AccessControlPolicy>

**URI Request Parameters**

The request uses the following URI parameters.

**Bucket (p. 355)**

The bucket name that contains the object to which you want to attach the ACL.

When using this API with an access point, you must direct requests to the access point hostname. The access point hostname takes the form `AccessPointName-AccountId.s3-accesspoint.Region.amazonaws.com`. When using this operation with an access point through the AWS SDKs, you provide the access point ARN in place of the bucket name. For more information about access point ARNs, see Using Access Points in the *Amazon Simple Storage Service Developer Guide*.

Required: Yes

**Content-MD5 (p. 355)**

*This header has been deprecated.*

The base64-encoded 128-bit MD5 digest of the data. This header must be used as a message integrity check to verify that the request body was not corrupted in transit. For more information, go to RFC 1864.>

**Key (p. 355)**

Key for which the PUT operation was initiated.

When using this API with an access point, you must direct requests to the access point hostname. The access point hostname takes the form `AccessPointName-AccountId.s3-accesspoint.Region.amazonaws.com`. When using this operation with an access point through the AWS SDKs, you provide the access point ARN in place of the bucket name. For more information about access point ARNs, see Using Access Points in the *Amazon Simple Storage Service Developer Guide*.

When using this API with Amazon S3 on Outposts, you must direct requests to the S3 on Outposts hostname. The S3 on Outposts hostname takes the form `AccessPointName-AccountId.outpostId.s3-
outposts.Region.amazonaws.com. When using this operation using S3 on Outposts through the AWS SDKs, you provide the Outposts bucket ARN in place of the bucket name. For more information about S3 on Outposts ARNs, see Using S3 on Outposts in the Amazon Simple Storage Service Developer Guide.

Length Constraints: Minimum length of 1.

Required: Yes

versionId (p. 355)

VersionId used to reference a specific version of the object.

x-amz-acl (p. 355)

The canned ACL to apply to the object. For more information, see Canned ACL.

Valid Values: private | public-read | public-read-write | authenticated-read | aws-exec-read | bucket-owner-read | bucket-owner-full-control

x-amz-expected-bucket-owner (p. 355)

The account id of the expected bucket owner. If the bucket is owned by a different account, the request will fail with an HTTP 403 (Access Denied) error.

x-amz-grant-full-control (p. 355)

Allows grantee the read, write, read ACP, and write ACP permissions on the bucket.

This action is not supported by Amazon S3 on Outposts.

x-amz-grant-read (p. 355)

Allows grantee to list the objects in the bucket.

This action is not supported by Amazon S3 on Outposts.

x-amz-grant-read-acp (p. 355)

Allows grantee to read the bucket ACL.

This action is not supported by Amazon S3 on Outposts.

x-amz-grant-write (p. 355)

Allows grantee to create, overwrite, and delete any object in the bucket.

x-amz-grant-write-acp (p. 355)

Allows grantee to write the ACL for the applicable bucket.

This action is not supported by Amazon S3 on Outposts.

x-amz-request-payer (p. 355)

Confirms that the requester knows that they will be charged for the request. Bucket owners need not specify this parameter in their requests. For information about downloading objects from requester pays buckets, see Downloading Objects in Requestor Pays Buckets in the Amazon S3 Developer Guide.

Valid Values: requester

Request Body

The request accepts the following data in XML format.
AccessControlPolicy (p. 355)

Root level tag for the AccessControlPolicy parameters.

Required: Yes

Grants (p. 355)

A list of grants.

Type: Array of Grant (p. 566) data types

Required: No

Owner (p. 355)

Container for the bucket owner's display name and ID.

Type: Owner (p. 617) data type

Required: No

Response Syntax

HTTP/1.1 200
x-amz-request-charged: RequestCharged

Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The response returns the following HTTP headers.

x-amz-request-charged (p. 358)

If present, indicates that the requester was successfully charged for the request.

Valid Values: requester

The specified key does not exist.

Examples

Sample Request

The following request grants access permission to an existing object. The request specifies the ACL in the body. In addition to granting full control to the object owner, the XML specifies full control to an AWS account identified by its canonical user ID.

```
PUT /my-image.jpg?acl HTTP/1.1
Host: bucket.s3.<Region>.amazonaws.com
Date: Wed, 28 Oct 2009 22:32:00 GMT
Authorization: authorization string
Content-Length: 124

<AccessControlPolicy>
  <Owner>
    <ID>75aa57f09aa0c8caebab4f8c24e99d10f8e7faeebf76c078efc7c6caa54ba06a</ID>
    <DisplayName>CustomersName@amazon.com</DisplayName>
  </Owner>
</AccessControlPolicy>
```
Sample Response

The following shows a sample response when versioning on the bucket is enabled.

HTTP/1.1 200 OK
x-amz-id-2: eftixk72aD6Ap51T9AS1ed4OpIsrzj7UDNEHGran
x-amz-request-id: 318BC8BC148832E5
x-amz-version-id: 3/L4kqtJlcpXrof3vJVBH40Nr6XgRDQBPUMLUo
Date: Wed, 28 Oct 2009 22:32:00 GMT
Last-Modified: Sun, 1 Jan 2006 12:00:00 GMT
Content-Length: 0
Connection: close
Server: AmazonS3

Sample Request: Setting the ACL of a specified object version

The following request sets the ACL on the specified version of the object.

PUT /my-image.jpg?acl&versionId=3HL4kqtJlcpXroDTdmJ+rmSpXd3dIbrHY+MTRCxf3vJVBH40Nrjfkd
HTTP/1.1
Host: bucket.s3.<Region>.amazonaws.com
Date: Wed, 28 Oct 2009 22:32:00 GMT
Authorization: authorization string
Content-Length: 124

<AccessControlPolicy>
  <Owner>
    <ID>75aa57f09aa0c8caeb4f8c24e99d10f8e7faeebf76c078efc7c6caea54ba06a</ID>
    <DisplayName>mtd@amazon.com</DisplayName>
  </Owner>
  <AccessControlList>
    <Grant>
      <Grantee xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:type="CanonicalUser">
        <ID>75aa57f09aa0c8caeb4f8c24e99d10f8e7faeebf76c078efc7c6caea54ba06a</ID>
        <DisplayName>mtd@amazon.com</DisplayName>
      </Grantee>
      <Permission>FULL_CONTROL</Permission>
    </Grant>
  </AccessControlList>
</AccessControlPolicy>

Sample Response

This example illustrates one usage of PutObjectAcl.
Sample Request: Access permissions specified using headers

The following request sets the ACL on the specified version of the object.

PUT ExampleObject.txt?acl HTTP/1.1
Host: examplebucket.s3.<Region>.amazonaws.com
x-amz-acl: public-read
Accept: */*
Authorization: authorization string
Host: s3.amazonaws.com
Connection: Keep-Alive

Sample Response

This example illustrates one usage of PutObjectAcl.

HTTP/1.1 200 OK
x-amz-id-2: w5YegkbG6ZDsje4WK56RWPxNQHIQ0CjrjyRVFZhEJI9E3kbabXnBO9w5G7Dmxs8gk
x-amz-request-id: C13B2827BD8455B1
Date: Sun, 29 Apr 2012 23:24:12 GMT
Content-Length: 0
Server: AmazonS3

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3
**PutObjectLegalHold**  
Service: Amazon Simple Storage Service  

Applies a Legal Hold configuration to the specified object.

This action is not supported by Amazon S3 on Outposts.

**Related Resources**
- Locking Objects

**Request Syntax**

```
PUT /{Key+}?legal-hold&versionId=VersionId HTTP/1.1
Host: Bucket.s3.amazonaws.com
x-amz-request-payer: RequestPayer
Content-MD5: ContentMD5
x-amz-expected-bucket-owner: ExpectedBucketOwner
<?xml version="1.0" encoding="UTF-8"?><LegalHold xmlns="http://s3.amazonaws.com/doc/2006-03-01/"/>
<Status>string</Status>
</LegalHold>
```

**URI Request Parameters**

The request uses the following URI parameters.

**Bucket (p. 361)**

The bucket name containing the object that you want to place a Legal Hold on.

When using this API with an access point, you must direct requests to the access point hostname. The access point hostname takes the form `AccessPointName-AccountId.s3-accesspoint.Region.amazonaws.com`. When using this operation with an access point through the AWS SDKs, you provide the access point ARN in place of the bucket name. For more information about access point ARNs, see Using Access Points in the Amazon Simple Storage Service Developer Guide.

Required: Yes

**Content-MD5 (p. 361)**

This header has been deprecated.

The MD5 hash for the request body.

**Key (p. 361)**

The key name for the object that you want to place a Legal Hold on.

Length Constraints: Minimum length of 1.

Required: Yes

**versionId (p. 361)**

The version ID of the object that you want to place a Legal Hold on.

The account id of the expected bucket owner. If the bucket is owned by a different account, the request will fail with an HTTP 403 (Access Denied) error.
x-amz-request-payer (p. 361)

Confirms that the requester knows that they will be charged for the request. Bucket owners need not specify this parameter in their requests. For information about downloading objects from requester pays buckets, see Downloading Objects in Requestor Pays Buckets in the Amazon S3 Developer Guide.

Valid Values: requester

Request Body

The request accepts the following data in XML format.

LegalHold (p. 361)

Root level tag for the LegalHold parameters.

Required: Yes

Status (p. 361)

Indicates whether the specified object has a Legal Hold in place.

Type: String

Valid Values: ON | OFF

Required: No

Response Syntax

```
HTTP/1.1 200
x-amz-request-charged: RequestCharged
```

Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The response returns the following HTTP headers.

x-amz-request-charged (p. 362)

If present, indicates that the requester was successfully charged for the request.

Valid Values: requester

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3
**PutObjectLockConfiguration**  
*Service: Amazon Simple Storage Service*

Places an Object Lock configuration on the specified bucket. The rule specified in the Object Lock configuration will be applied by default to every new object placed in the specified bucket.

**Note**

DefaultRetention requires either Days or Years. You can't specify both at the same time.

**Related Resources**

- Locking Objects

**Request Syntax**

```xml
PUT /?object-lock HTTP/1.1  
Host: Bucket.s3.amazonaws.com  
x-amz-request-payer: RequestPayer  
x-amz-bucket-object-lock-token: Token  
Content-MD5: ContentMD5  
<xamz-expected-bucket-owner: ExpectedBucketOwner>
<xml version="1.0" encoding="UTF-8"?>
  <ObjectLockEnabled>string</ObjectLockEnabled>
  <Rule>
    <DefaultRetention>
      <Days>integer</Days>
      <Mode>string</Mode>
      <Years>integer</Years>
    </DefaultRetention>
  </Rule>
</ObjectLockConfiguration>
```

**URI Request Parameters**

The request uses the following URI parameters.

**Bucket (p. 364)**

The bucket whose Object Lock configuration you want to create or replace.

- Required: Yes

**Content-MD5 (p. 364)**

*This header has been deprecated.*

The MD5 hash for the request body.

**x-amz-bucket-object-lock-token (p. 364)**

A token to allow Object Lock to be enabled for an existing bucket.

**x-amz-expected-bucket-owner (p. 364)**

The account id of the expected bucket owner. If the bucket is owned by a different account, the request will fail with an HTTP 403 (Access Denied) error.

**x-amz-request-payer (p. 364)**

Confirms that the requester knows that they will be charged for the request. Bucket owners need not specify this parameter in their requests. For information about downloading objects from
Valid Values: requester

Request Body

The request accepts the following data in XML format.

ObjectLockConfiguration (p. 364)

Root level tag for the ObjectLockConfiguration parameters.

Required: Yes

ObjectLockEnabled (p. 364)

Indicates whether this bucket has an Object Lock configuration enabled.

Type: String

Valid Values: Enabled

Required: No

Rule (p. 364)

The Object Lock rule in place for the specified object.

Type: ObjectLockRule (p. 612) data type

Required: No

Response Syntax

HTTP/1.1 200
x-amz-request-charged: RequestCharged

Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The response returns the following HTTP headers.

x-amz-request-charged (p. 365)

If present, indicates that the requester was successfully charged for the request.

Valid Values: requester

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go

API Version 2006-03-01
- AWS SDK for Java
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3
**PutObjectRetention**

Service: Amazon Simple Storage Service

Places an Object Retention configuration on an object.

This action is not supported by Amazon S3 on Outposts.

**Related Resources**

- Locking Objects

**Request Syntax**

```
PUT /{Key+}?retention&versionId=VersionId HTTP/1.1
Host: Bucket.s3.amazonaws.com
x-amz-request-payer: RequestPayer
x-amz-bypass-governance-retention: BypassGovernanceRetention
Content-MD5: ContentMD5
x-amz-expected-bucket-owner: ExpectedBucketOwner
<?xml version="1.0" encoding="UTF-8"?>
<Retention xmlns="http://s3.amazonaws.com/doc/2006-03-01/">
  <Mode>string</Mode>
  <RetainUntilDate>timestamp</RetainUntilDate>
</Retention>
```

**URI Request Parameters**

The request uses the following URI parameters.

**Bucket (p. 367)**

The bucket name that contains the object you want to apply this Object Retention configuration to.

When using this API with an access point, you must direct requests to the access point hostname. The access point hostname takes the form `AccessPointName-AccountId.s3-accesspoint.Region.amazonaws.com`. When using this operation with an access point through the AWS SDKs, you provide the access point ARN in place of the bucket name. For more information about access point ARNs, see Using Access Points in the Amazon Simple Storage Service Developer Guide.

Required: Yes

**Content-MD5 (p. 367)**

*This header has been deprecated.*

The MD5 hash for the request body.

**Key (p. 367)**

The key name for the object that you want to apply this Object Retention configuration to.

Length Constraints: Minimum length of 1.

Required: Yes

**versionId (p. 367)**

The version ID for the object that you want to apply this Object Retention configuration to.
**x-amz-bypass-governance-retention (p. 367)**

Indicates whether this operation should bypass Governance-mode restrictions.

**x-amz-expected-bucket-owner (p. 367)**

The account id of the expected bucket owner. If the bucket is owned by a different account, the request will fail with an HTTP 403 (Access Denied) error.

**x-amz-request-payer (p. 367)**

Confirms that the requester knows that they will be charged for the request. Bucket owners need not specify this parameter in their requests. For information about downloading objects from requester pays buckets, see Downloading Objects in Requestor Pays Buckets in the Amazon S3 Developer Guide.

Valid Values: requester

**Request Body**

The request accepts the following data in XML format.

**Retention (p. 367)**

Root level tag for the Retention parameters.

Required: Yes

**Mode (p. 367)**

Indicates the Retention mode for the specified object.

Type: String

Valid Values: GOVERNANCE | COMPLIANCE

Required: No

**RetainUntilDate (p. 367)**

The date on which this Object Lock Retention will expire.

Type: Timestamp

Required: No

**Response Syntax**

```
HTTP/1.1 200
x-amz-request-charged: RequestCharged
```

**Response Elements**

If the action is successful, the service sends back an HTTP 200 response.

The response returns the following HTTP headers.

**x-amz-request-charged (p. 368)**

If present, indicates that the requester was successfully charged for the request.
Valid Values: requester

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3
PutObjectTagging
Service: Amazon Simple Storage Service

Sets the supplied tag-set to an object that already exists in a bucket.

A tag is a key-value pair. You can associate tags with an object by sending a PUT request against the tagging subresource that is associated with the object. You can retrieve tags by sending a GET request. For more information, see GetObjectTagging.

For tagging-related restrictions related to characters and encodings, see Tag Restrictions. Note that Amazon S3 limits the maximum number of tags to 10 tags per object.

To use this operation, you must have permission to perform the s3:PutObjectTagging action. By default, the bucket owner has this permission and can grant this permission to others.

To put tags of any other version, use the versionId query parameter. You also need permission for the s3:PutObjectVersionTagging action.

For information about the Amazon S3 object tagging feature, see Object Tagging.

Special Errors

- Code: InvalidTagError
  Cause: The tag provided was not a valid tag. This error can occur if the tag did not pass input validation. For more information, see Object Tagging.

- Code: MalformedXMLError
  Cause: The XML provided does not match the schema.

- Code: OperationAbortedError
  Cause: A conflicting conditional operation is currently in progress against this resource. Please try again.

- Code: InternalError
  Cause: The service was unable to apply the provided tag to the object.

Related Resources

- GetObjectTagging

Request Syntax

PUT /{Key+}?tagging&versionId=VersionId HTTP/1.1
Host: Bucket.s3.amazonaws.com
Content-MD5: ContentMD5
x-amz-expected-bucket-owner: ExpectedBucketOwner
<?xml version="1.0" encoding="UTF-8"?>
<Tagging xmlns="http://s3.amazonaws.com/doc/2006-03-01/">
  <TagSet>
    <Tag>
      <Key>string</Key>
      <Value>string</Value>
    </Tag>
  </TagSet>
</Tagging>

URI Request Parameters

The request uses the following URI parameters.
Bucket (p. 370)

The bucket name containing the object.

When using this API with an access point, you must direct requests to the access point hostname. The access point hostname takes the form AccessPointName-AccountId.s3-accesspoint.Region.amazonaws.com. When using this operation with an access point through the AWS SDKs, you provide the access point ARN in place of the bucket name. For more information about access point ARNs, see Using Access Points in the Amazon Simple Storage Service Developer Guide.

When using this API with Amazon S3 on Outposts, you must direct requests to the S3 on Outposts hostname. The S3 on Outposts hostname takes the form AccessPointName-AccountId.outpostId.s3-outposts.Region.amazonaws.com. When using this operation using S3 on Outposts through the AWS SDKs, you provide the Outposts bucket ARN in place of the bucket name. For more information about S3 on Outposts ARNs, see Using S3 on Outposts in the Amazon Simple Storage Service Developer Guide.

Required: Yes

Content-MD5 (p. 370)

This header has been deprecated.

The MD5 hash for the request body.

Key (p. 370)

Name of the object key.

Length Constraints: Minimum length of 1.

Required: Yes

VersionId (p. 370)

The versionId of the object that the tag-set will be added to.

x-amz-expected-bucket-owner (p. 370)

The account id of the expected bucket owner. If the bucket is owned by a different account, the request will fail with an HTTP 403 (Access Denied) error.

Request Body

The request accepts the following data in XML format.

Tagging (p. 370)

Root level tag for the Tagging parameters.

Required: Yes

TagSet (p. 370)

A collection for a set of tags

Type: Array of Tag (p. 666) data types

Required: Yes

Response Syntax

HTTP/1.1 200
Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The response returns the following HTTP headers.

**x-amz-version-id (p. 371)**

The versionId of the object the tag-set was added to.

Examples

Sample Request: Add tag set to an object

The following request adds a tag set to the existing object object-key in the examplebucket bucket.

```
PUT object-key?tagging HTTP/1.1
Host: examplebucket.s3.<Region>.amazonaws.com
Content-Length: length
Content-MD5: pUNXr/BjKK5G2UKExample==
x-amz-date: 20160923T001956Z
Authorization: authorization string
<Tagging>
  <TagSet>
    <Tag>
      <Key>tag1</Key>
      <Value>val1</Value>
    </Tag>
    <Tag>
      <Key>tag2</Key>
      <Value>val2</Value>
    </Tag>
  </TagSet>
</Tagging>
```

Sample Response

This example illustrates one usage of PutObjectTagging.

```
HTTP/1.1 200 OK
x-amz-id-2: YgIPIfBiKa2bj0KMgUAdQkf3ShJTOOpXUueF6QKo
x-amz-request-id: 236A8905248E5A01
Date: Fri, 23 Sep 2016 00:20:19 GMT
```

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
• AWS SDK for Java
• AWS SDK for JavaScript
• AWS SDK for PHP V3
• AWS SDK for Python
• AWS SDK for Ruby V3
**PutPublicAccessBlock**  
Service: Amazon Simple Storage Service

Creates or modifies the PublicAccessBlock configuration for an Amazon S3 bucket. To use this operation, you must have the `s3:PutBucketPublicAccessBlock` permission. For more information about Amazon S3 permissions, see [Specifying Permissions in a Policy](#).

**Important**  
When Amazon S3 evaluates the PublicAccessBlock configuration for a bucket or an object, it checks the PublicAccessBlock configuration for both the bucket (or the bucket that contains the object) and the bucket owner's account. If the PublicAccessBlock configurations are different between the bucket and the account, Amazon S3 uses the most restrictive combination of the bucket-level and account-level settings.

For more information about when Amazon S3 considers a bucket or an object public, see [The Meaning of "Public"](#).

**Related Resources**
- GetPublicAccessBlock
- DeletePublicAccessBlock
- GetBucketPolicyStatus
- Using Amazon S3 Block Public Access

**Request Syntax**

```xml
PUT /?publicAccessBlock HTTP/1.1  
Host: Bucket.s3.amazonaws.com  
Content-MD5: ContentMD5  
x-amz-expected-bucket-owner: ExpectedBucketOwner  
<?xml version="1.0" encoding="UTF-8"?>  
   <BlockPublicAcls>boolean</BlockPublicAcls>  
   <IgnorePublicAcls>boolean</IgnorePublicAcls>  
   <BlockPublicPolicy>boolean</BlockPublicPolicy>  
   <RestrictPublicBuckets>boolean</RestrictPublicBuckets>  
</PublicAccessBlockConfiguration>
```

**URI Request Parameters**

The request uses the following URI parameters.

**Bucket (p. 374)**

The name of the Amazon S3 bucket whose PublicAccessBlock configuration you want to set.

Required: Yes  

**Content-MD5 (p. 374)**

*This header has been deprecated.*

The MD5 hash of the PutPublicAccessBlock request body.

**x-amz-expected-bucket-owner (p. 374)**

The account id of the expected bucket owner. If the bucket is owned by a different account, the request will fail with an HTTP 403 (Access Denied) error.
Request Body

The request accepts the following data in XML format.

PublicAccessBlockConfiguration (p. 374)

Root level tag for the PublicAccessBlockConfiguration parameters.
Required: Yes

BlockPublicAcls (p. 374)

Specifies whether Amazon S3 should block public access control lists (ACLs) for this bucket and objects in this bucket. Setting this element to TRUE causes the following behavior:
• PUT Bucket acl and PUT Object acl calls fail if the specified ACL is public.
• PUT Object calls fail if the request includes a public ACL.
• PUT Bucket calls fail if the request includes a public ACL.

Enabling this setting doesn't affect existing policies or ACLs.
Type: Boolean
Required: No

BlockPublicPolicy (p. 374)

Specifies whether Amazon S3 should block public bucket policies for this bucket. Setting this element to TRUE causes Amazon S3 to reject calls to PUT Bucket policy if the specified bucket policy allows public access.

Enabling this setting doesn't affect existing bucket policies.
Type: Boolean
Required: No

IgnorePublicAcls (p. 374)

Specifies whether Amazon S3 should ignore public ACLs for this bucket and objects in this bucket. Setting this element to TRUE causes Amazon S3 to ignore all public ACLs on this bucket and objects in this bucket.

Enabling this setting doesn't affect the persistence of any existing ACLs and doesn't prevent new public ACLs from being set.
Type: Boolean
Required: No

RestrictPublicBuckets (p. 374)

Specifies whether Amazon S3 should restrict public bucket policies for this bucket. Setting this element to TRUE restricts access to this bucket to only AWS service principals and authorized users within this account if the bucket has a public policy.

Enabling this setting doesn't affect previously stored bucket policies, except that public and cross-account access within any public bucket policy, including non-public delegation to specific accounts, is blocked.
Type: Boolean
Required: No
Response Syntax

HTTP/1.1 200

Response Elements

If the action is successful, the service sends back an HTTP 200 response with an empty HTTP body.

Examples

First Sample Request

The following request puts a bucket `PublicAccessBlock` configuration that rejects public ACLs.

```plaintext
PUT /<bucket-name>?publicAccessBlock HTTP/1.1
Host: <bucket-name>.s3.<Region>.amazonaws.com
x-amz-date: <Thu, 15 Nov 2016 00:17:21 GMT>
Authorization: <signatureValue>

<?xml version="1.0" encoding="UTF-8"?>
<PublicAccessBlockConfiguration>
  <BlockPublicAcls>TRUE</BlockPublicAcls>
  <IgnorePublicAcls>FALSE</IgnorePublicAcls>
  <BlockPublicPolicy>FALSE</BlockPublicPolicy>
  <RestrictPublicBuckets>FALSE</RestrictPublicBuckets>
</PublicAccessBlockConfiguration>
```

First Sample Response

This example illustrates one usage of `PutPublicAccessBlock`.

```plaintext
HTTP/1.1 200 OK
x-amz-id-2: ITnGT1y4REXAMPLEPi4hklTXouTf0hccUjo0iCPEXAMPLEutBj3M7fPGlWO2SEWp
x-amz-request-id: 51991EXAMPLE5321
Date: Thu, 15 Nov 2016 00:17:22 GMT
Server: AmazonS3
Content-Length: 0
```

Second Sample Request

The following request puts a bucket `PublicAccessBlock` configuration that ignores public ACLs and restricts access to public buckets.

```plaintext
PUT /<bucket-name>?publicAccessBlock HTTP/1.1
Host: <bucket-name>.s3.<Region>.amazonaws.com
x-amz-date: <Thu, 15 Nov 2016 00:17:21 GMT>
Authorization: <signatureValue>

<?xml version="1.0" encoding="UTF-8"?>
<PublicAccessBlockConfiguration>
  <BlockPublicAcls>FALSE</BlockPublicAcls>
  <IgnorePublicAcls>TRUE</IgnorePublicAcls>
  <BlockPublicPolicy>FALSE</BlockPublicPolicy>
  <RestrictPublicBuckets>TRUE</RestrictPublicBuckets>
</PublicAccessBlockConfiguration>
```
Second Sample Response

This example illustrates one usage of PutPublicAccessBlock.

HTTP/1.1 200 OK
x-amz-id-2: ITnGTly4REXAMPLEPi4hklTXouTf0hccUjoOicFEXAMPLEutBj3M7fPGlWO2SEWp
x-amz-request-id: 51991EXAMPLE5321
Date: Thu, 15 Nov 2016 00:17:22 GMT
Server: AmazonS3
Content-Length: 0

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3
RestoreObject
Service: Amazon Simple Storage Service

Restores an archived copy of an object back into Amazon S3

This action is not supported by Amazon S3 on Outposts.

This action performs the following types of requests:

• select - Perform a select query on an archived object
• restore an archive - Restore an archived object

To use this operation, you must have permissions to perform the s3:RestoreObject action. The bucket owner has this permission by default and can grant this permission to others. For more information about permissions, see Permissions Related to Bucket Subresource Operations and Managing Access Permissions to Your Amazon S3 Resources in the Amazon Simple Storage Service Developer Guide.

Querying Archives with Select Requests

You use a select type of request to perform SQL queries on archived objects. The archived objects that are being queried by the select request must be formatted as uncompressed comma-separated values (CSV) files. You can run queries and custom analytics on your archived data without having to restore your data to a hotter Amazon S3 tier. For an overview about select requests, see Querying Archived Objects in the Amazon Simple Storage Service Developer Guide.

When making a select request, do the following:

• Define an output location for the select query's output. This must be an Amazon S3 bucket in the same AWS Region as the bucket that contains the archive object that is being queried. The AWS account that initiates the job must have permissions to write to the S3 bucket. You can specify the storage class and encryption for the output objects stored in the bucket. For more information about output, see Querying Archived Objects in the Amazon Simple Storage Service Developer Guide.

For more information about the S3 structure in the request body, see the following:

• PutObject
• Managing Access with ACLs in the Amazon Simple Storage Service Developer Guide
• Protecting Data Using Server-Side Encryption in the Amazon Simple Storage Service Developer Guide
• Define the SQL expression for the SELECT type of restoration for your query in the request body's SelectParameters structure. You can use expressions like the following examples.
• The following expression returns all records from the specified object.

  SELECT * FROM Object

• Assuming that you are not using any headers for data stored in the object, you can specify columns with positional headers.

  SELECT s._1, s._2 FROM Object s WHERE s._3 > 100

• If you have headers and you set the fileHeaderInfo in the CSV structure in the request body to USE, you can specify headers in the query. (If you set the fileHeaderInfo field to IGNORE, the first row is skipped for the query.) You cannot mix ordinal positions with header column names.

  SELECT s.Id, s.FirstName, s.SSN FROM S3Object s

For more information about using SQL with S3 Glacier Select restore, see SQL Reference for Amazon S3 Select and S3 Glacier Select in the Amazon Simple Storage Service Developer Guide.
When making a select request, you can also do the following:

- To expedite your queries, specify the **Expedited** tier. For more information about tiers, see "Restoring Archives," later in this topic.
- Specify details about the data serialization format of both the input object that is being queried and the serialization of the CSV-encoded query results.

The following are additional important facts about the select feature:

- The output results are new Amazon S3 objects. Unlike archive retrievals, they are stored until explicitly deleted—manually or through a lifecycle policy.
- You can issue more than one select request on the same Amazon S3 object. Amazon S3 doesn't deduplicate requests, so avoid issuing duplicate requests.
- Amazon S3 accepts a select request even if the object has already been restored. A select request doesn't return error response 409.

**Restoring Archives**

Objects that you archive to the S3 Glacier, S3 Glacier Deep Archive, S3 Intelligent-Tiering Archive, or S3 Intelligent-Tiering Deep Archive storage classes are not accessible in real time. For objects in Archive Access tier or Deep Archive Access tier you must first initiate a restore request, and then wait until the object is moved into the Frequent Access tier. For objects in S3 Glacier or S3 Glacier Deep Archive you must first initiate a restore request, and then wait until a temporary copy of the object is available. To access an archived object, you must restore the object for the duration (number of days) that you specify.

To restore a specific object version, you can provide a version ID. If you don't provide a version ID, Amazon S3 restores the current version.

When restoring an archived object (or using a select request), you can specify one of the following data access tier options in the **Tier** element of the request body:

- **Expedited** - Expedited retrievals allow you to quickly access your data stored in the S3 Glacier or S3 Intelligent-Tiering Archive storage class when occasional urgent requests for a subset of archives are required. For all but the largest archived objects (250 MB+), data accessed using Expedited retrievals is typically made available within 1–5 minutes. Provisioned capacity ensures that retrieval capacity for Expedited retrievals is available when you need it. Expedited retrievals and provisioned capacity are not available for objects stored in the S3 Glacier Deep Archive or S3 Intelligent-Tiering Deep Archive storage class.
- **Standard** - Standard retrievals allow you to access any of your archived objects within several hours. This is the default option for retrieval requests that do not specify the retrieval option. Standard retrievals typically finish within 3–5 hours for objects stored in the S3 Glacier or S3 Intelligent-Tiering Archive storage class. They typically finish within 12 hours for objects stored in the S3 Glacier Deep Archive or S3 Intelligent-Tiering Deep Archive storage class. Standard retrievals are free for objects stored in S3 Intelligent-Tiering.
- **Bulk** - Bulk retrievals are the lowest-cost retrieval option in S3 Glacier, enabling you to retrieve large amounts, even petabytes, of data inexpensively. Bulk retrievals typically finish within 5–12 hours for objects stored in the S3 Glacier or S3 Intelligent-Tiering Archive storage class. They typically finish within 48 hours for objects stored in the S3 Glacier Deep Archive or S3 Intelligent-Tiering Deep Archive storage class. Bulk retrievals are free for objects stored in S3 Intelligent-Tiering.

For more information about archive retrieval options and provisioned capacity for **Expedited** data access, see **Restoring Archived Objects** in the *Amazon Simple Storage Service Developer Guide*. 
You can use Amazon S3 restore speed upgrade to change the restore speed to a faster speed while it is in progress. For more information, see Upgrading the speed of an in-progress restore in the Amazon Simple Storage Service Developer Guide.

To get the status of object restoration, you can send a HEAD request. Operations return the x-amz-restore header, which provides information about the restoration status, in the response. You can use Amazon S3 event notifications to notify you when a restore is initiated or completed. For more information, see Configuring Amazon S3 Event Notifications in the Amazon Simple Storage Service Developer Guide.

After restoring an archived object, you can update the restoration period by reissuing the request with a new period. Amazon S3 updates the restoration period relative to the current time and charges only for the request—there are no data transfer charges. You cannot update the restoration period when Amazon S3 is actively processing your current restore request for the object.

If your bucket has a lifecycle configuration with a rule that includes an expiration action, the object expiration overrides the life span that you specify in a restore request. For example, if you restore an object copy for 10 days, but the object is scheduled to expire in 3 days, Amazon S3 deletes the object in 3 days. For more information about lifecycle configuration, see PutBucketLifecycleConfiguration and Object Lifecycle Management in Amazon Simple Storage Service Developer Guide.

Responses

A successful operation returns either the 200 OK or 202 Accepted status code.

- If the object is not previously restored, then Amazon S3 returns 202 Accepted in the response.
- If the object is previously restored, Amazon S3 returns 200 OK in the response.

Special Errors

- Code: RestoreAlreadyInProgress
  Cause: Object restore is already in progress. (This error does not apply to SELECT type requests.)
  HTTP Status Code: 409 Conflict
  SOAP Fault Code Prefix: Client
- Code: GlacierExpeditedRetrievalNotAvailable
  Cause: expedited retrievals are currently not available. Try again later. (Returned if there is insufficient capacity to process the Expedited request. This error applies only to Expedited retrievals and not to S3 Standard or Bulk retrievals.)
  HTTP Status Code: 503
  SOAP Fault Code Prefix: N/A

Related Resources

- PutBucketLifecycleConfiguration
- GetBucketNotificationConfiguration
- SQL Reference for Amazon S3 Select and S3 Glacier Select in the Amazon Simple Storage Service Developer Guide

Request Syntax

```bash
POST /{Key+}?restore&versionId=VersionId HTTP/1.1
Host: Bucket.s3.amazonaws.com
x-amz-request-payer: RequestPayer
x-amz-expected-bucket-owner: ExpectedBucketOwner
```
<?xml version="1.0" encoding="UTF-8"?>
<RestoreRequest xmlns="http://s3.amazonaws.com/doc/2006-03-01/">
  <Days>integer</Days>
  <GlacierJobParameters>
    <Tier>string</Tier>
  </GlacierJobParameters>
  <Type>string</Type>
  <Tier>string</Tier>
  <Description>string</Description>
  <SelectParameters>
    <Expression>string</Expression>
    <ExpressionType>string</ExpressionType>
    <InputSerialization>
      <CompressionType>string</CompressionType>
      <CSV>
        <AllowQuotedRecordDelimiter>boolean</AllowQuotedRecordDelimiter>
        <Comments>string</Comments>
        <FieldDelimiter>string</FieldDelimiter>
        <FileHeaderInfo>string</FileHeaderInfo>
        <QuoteCharacter>string</QuoteCharacter>
        <QuoteEscapeCharacter>string</QuoteEscapeCharacter>
        <RecordDelimiter>string</RecordDelimiter>
      </CSV>
      <JSON>
      </JSON>
      <Parquet>
      </Parquet>
    </InputSerialization>
    <OutputSerialization>
      <CSV>
        <FieldDelimiter>string</FieldDelimiter>
        <QuoteCharacter>string</QuoteCharacter>
        <QuoteEscapeCharacter>string</QuoteEscapeCharacter>
        <QuoteFields>string</QuoteFields>
        <RecordDelimiter>string</RecordDelimiter>
      </CSV>
      <JSON>
        <RecordDelimiter>string</RecordDelimiter>
      </JSON>
      <Parquet>
      </Parquet>
    </OutputSerialization>
  </SelectParameters>
  <OutputLocation>
    <S3>
      <AccessControlList>
        <Grant>
          <Grantee>
            <DisplayName>string</DisplayName>
            <EmailAddress>string</EmailAddress>
            <ID>string</ID>
            <xsi:type>string</xsi:type>
            <URI>string</URI>
          </Grantee>
          <Permission>string</Permission>
        </Grant>
      </AccessControlList>
      <BucketName>string</BucketName>
      <CannedACL>string</CannedACL>
      <Encryption>
        <EncryptionType>string</EncryptionType>
        <KMSContext>string</KMSContext>
        <KMSKeyId>string</KMSKeyId>
      </Encryption>
      <Prefix>string</Prefix>
      <StorageClass>string</StorageClass>
      <Tagging>
    </S3>
  </OutputLocation>
</RestoreRequest>
<TagSet>
  <Tag>
    <Key>string</Key>
    <Value>string</Value>
  </Tag>
</TagSet>
</Tagging>
</UserMetadata>
</S3>
</OutputLocation>
</RestoreRequest>

**URI Request Parameters**

The request uses the following URI parameters.

**Bucket (p. 380)**

The bucket name containing the object to restore.

When using this API with an access point, you must direct requests to the access point hostname. The access point hostname takes the form `AccessPointName-AccountId.s3-accesspoint.Region.amazonaws.com`. When using this API with an access point through the AWS SDKs, you provide the access point ARN in place of the bucket name. For more information about access point ARNs, see Using Access Points in the Amazon Simple Storage Service Developer Guide.

When using this API with Amazon S3 on Outposts, you must direct requests to the S3 on Outposts hostname. The S3 on Outposts hostname takes the form `AccessPointName-AccountId.outpostID.s3-outposts.Region.amazonaws.com`. When using this operation using S3 on Outposts through the AWS SDKs, you provide the Outposts bucket ARN in place of the bucket name. For more information about S3 on Outposts ARNs, see Using S3 on Outposts in the Amazon Simple Storage Service Developer Guide.

Required: Yes

**Key (p. 380)**

Object key for which the operation was initiated.

Length Constraints: Minimum length of 1.

Required: Yes

**versionId (p. 380)**

VersionId used to reference a specific version of the object.

**x-amz-expected-bucket-owner (p. 380)**

The account id of the expected bucket owner. If the bucket is owned by a different account, the request will fail with an HTTP 403 (Access Denied) error.

**x-amz-request-payer (p. 380)**

Confirms that the requester knows that they will be charged for the request. Bucket owners need not specify this parameter in their requests. For information about downloading objects from requester pays buckets, see Downloading Objects in Requestor Pays Buckets in the Amazon S3 Developer Guide.
Valid Values: requester

Request Body

The request accepts the following data in XML format.

**RestoreRequest (p. 380)**

Root level tag for the RestoreRequest parameters.

**Days (p. 380)**

Lifetime of the active copy in days. Do not use with restores that specify OutputLocation.

The Days element is required for regular restores, and must not be provided for select requests.

Type: Integer

**Description (p. 380)**

The optional description for the job.

Type: String

**GlacierJobParameters (p. 380)**

S3 Glacier related parameters pertaining to this job. Do not use with restores that specify OutputLocation.

Type: GlacierJobParameters (p. 565) data type

**OutputLocation (p. 380)**

Describes the location where the restore job's output is stored.

Type: OutputLocation (p. 615) data type

**SelectParameters (p. 380)**

Describes the parameters for Select job types.

Type: SelectParameters (p. 654) data type

**Tier (p. 380)**

Retrieval tier at which the restore will be processed.

Type: String

Valid Values: Standard | Bulk | Expedited

Required: No
Type of restore request.

Type: String

Valid Values: SELECT

Required: No

Response Syntax

HTTP/1.1 200
x-amz-request-charged: RequestCharged
x-amz-restore-output-path: RestoreOutputPath

Response Elements

If the action is successful, the service sends back an HTTP 200 response. The response returns the following HTTP headers.

x-amz-request-charged (p. 384)

If present, indicates that the requester was successfully charged for the request.

Valid Values: requester

x-amz-restore-output-path (p. 384)

Indicates the path in the provided S3 output location where Select results will be restored to.

This operation is not allowed against this storage tier.

Examples

Restore an object for 2 days using the expedited retrieval option

The following restore request restores a copy of the photo1.jpg object from S3 Glacier for a period of two days using the expedited retrieval option.

Host: examplebucket.dummy value
Date: Mon, 22 Oct 2012 01:49:52 GMT
Authorization: authorization string
Content-Length: content length

<RestoreRequest>
  <Days>2</Days>
  <GlacierJobParameters>
    <Tier>Expedited</Tier>
  </GlacierJobParameters>
</RestoreRequest>

Example

If the examplebucket does not have a restored copy of the object, Amazon S3 returns the following 202 Accepted response.
Note
If a copy of the object is already restored, Amazon S3 returns a 200 OK response, and updates only the restored copy's expiry time.

HTTP/1.1 202 Accepted
x-amz-id-2: GFihv3y6+kE7KG11GkqhU7/2/chR3Yb2fCb2S04nxI423Dqwg2XiQ0B/UZlzYQvPiBlZNRcovw=
x-amz-request-id: 9F341CD3C4BA79E0
Date: Sat, 20 Oct 2012 23:54:05 GMT
Content-Length: 0
Server: AmazonS3

Query an archive with a SELECT request

The following is an example SELECT restore request.

POST /object-one.csv?restore HTTP/1.1
Host: examplebucket.dummy value
Date: Date: Sat, 20 Oct 2012 23:54:05 GMT
Authorization: <>authorization string</>
Content-Length: <>content length</>

<RestoreRequest xmlns="http://s3.amazonaws.com/doc/2006-03-01/">
  <Type>SELECT</Type>
  <Tier>Expedited</Tier>
  <Description>this is a description</Description>
  <SelectParameters>
    <InputSerialization>
      <CSV>
        <FileHeaderInfo>IGNORE</FileHeaderInfo>
        <Comments>#$</Comments>
        <QuoteEscapeCharacter>"</QuoteEscapeCharacter>
        <RecordDelimiter>
          <FieldDelimiter>,</FieldDelimiter>
          <QuoteCharacter>"</QuoteCharacter>
        </RecordDelimiter>
      </CSV>
    </InputSerialization>
    <ExpressionType>SQL</ExpressionType>
    <Expression>select * from object</Expression>
    <OutputSerialization>
      <CSV>
        <QuoteFields>ALWAYS</QuoteFields>
        <QuoteEscapeCharacter>"</QuoteEscapeCharacter>
        <RecordDelimiter>
          <FieldDelimiter>\</FieldDelimiter>
          <QuoteCharacter>\</QuoteCharacter>
        </RecordDelimiter>
      </CSV>
    </OutputSerialization>
  </SelectParameters>
  <OutputLocation>
    <S3>
      <BucketName>example-output-bucket</BucketName>
      <Prefix>test-s3</Prefix>
      <AccessControlList>
        <Grant>
          <Grantee xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
            xsi:type="AmazonCustomerByEmail">
              <EmailAddress>jane-doe@example.com</EmailAddress>
            </Grantee>
            <Permission>FULL_CONTROL</Permission>
          </Grant>
        </AccessControlList>
      </S3>
    </OutputLocation>
  </RestoreRequest>
Example

Amazon S3 returns the following 202 Accepted response.

```
HTTP/1.1 202 Accepted
x-amz-id-2: GFihv3y6<k87KG11GEkQhU7/2/chR3Yb2fcb2S04nxI423Dqwg2XiQ0B/UZ1zYvPiBlZNRCovw=
          x-amz-request-id: 9F341CD3C4BA79E0
          x-amz-restore-output-path: js-test-s3/qE8nkSM0XIj-LuZE2HXXw6empQm3znLkH1MWInRYPs-
                                      OrjLW0uj6LyYm-neTvml-btz3wbBxfMhPykd3jkl-1vZE7w42/
          Date: Sat, 20 Oct 2012 23:54:05 GMT
          Content-Length: 0
          Server: AmazonS3
```

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3
SelectObjectContent
Service: Amazon Simple Storage Service

This operation filters the contents of an Amazon S3 object based on a simple structured query language (SQL) statement. In the request, along with the SQL expression, you must also specify a data serialization format (JSON, CSV, or Apache Parquet) of the object. Amazon S3 uses this format to parse object data into records, and returns only records that match the specified SQL expression. You must also specify the data serialization format for the response.

This action is not supported by Amazon S3 on Outposts.

For more information about Amazon S3 Select, see Selecting Content from Objects in the Amazon Simple Storage Service Developer Guide.

For more information about using SQL with Amazon S3 Select, see SQL Reference for Amazon S3 Select and S3 Glacier Select in the Amazon Simple Storage Service Developer Guide.

Permissions
You must have s3:GetObject permission for this operation. Amazon S3 Select does not support anonymous access. For more information about permissions, see Specifying Permissions in a Policy in the Amazon Simple Storage Service Developer Guide.

Object Data Formats
You can use Amazon S3 Select to query objects that have the following format properties:

- **CSV, JSON, and Parquet** - Objects must be in CSV, JSON, or Parquet format.
- **UTF-8** - UTF-8 is the only encoding type Amazon S3 Select supports.
- **GZIP or BZIP2** - CSV and JSON files can be compressed using GZIP or BZIP2. GZIP and BZIP2 are the only compression formats that Amazon S3 Select supports for CSV and JSON files. Amazon S3 Select supports columnar compression for Parquet using GZIP or Snappy. Amazon S3 Select does not support whole-object compression for Parquet objects.
- **Server-side encryption** - Amazon S3 Select supports querying objects that are protected with server-side encryption.

For objects that are encrypted with customer-provided encryption keys (SSE-C), you must use HTTPS, and you must use the headers that are documented in the GetObject. For more information about SSE-C, see Server-Side Encryption (Using Customer-Provided Encryption Keys) in the Amazon Simple Storage Service Developer Guide.

For objects that are encrypted with Amazon S3 managed encryption keys (SSE-S3) and customer master keys (CMKs) stored in AWS Key Management Service (SSE-KMS), server-side encryption is handled transparently, so you don't need to specify anything. For more information about server-side encryption, including SSE-S3 and SSE-KMS, see Protecting Data Using Server-Side Encryption in the Amazon Simple Storage Service Developer Guide.

Working with the Response Body

Given the response size is unknown, Amazon S3 Select streams the response as a series of messages and includes a Transfer-Encoding header with chunked as its value in the response. For more information, see Appendix: SelectObjectContent Response.

GetObject Support

The SelectObjectContent operation does not support the following GetObject functionality. For more information, see GetObject.
• Range: Although you can specify a scan range for an Amazon S3 Select request (see SelectObjectContentRequest - ScanRange in the request parameters), you cannot specify the range of bytes of an object to return.

• GLACIER, DEEP_ARCHIVE and REDUCED_REDUNDANCY storage classes: You cannot specify the GLACIER, DEEP_ARCHIVE, or REDUCED_REDUNDANCY storage classes. For more information, about storage classes see Storage Classes in the Amazon Simple Storage Service Developer Guide.

Special Errors

For a list of special errors for this operation, see List of SELECT Object Content Error Codes

Related Resources

• GetObject
• GetBucketLifecycleConfiguration
• PutBucketLifecycleConfiguration

Request Syntax

POST /{Key+}?select&select-type=2 HTTP/1.1
Host: Bucket.s3.amazonaws.com
x-amz-server-side-encryption-customer-algorithm: SSECustomerAlgorithm
x-amz-server-side-encryption-customer-key: SSECustomerKey
x-amz-server-side-encryption-customer-key-MD5: SSECustomerKeyMD5
x-amz-expected-bucket-owner: ExpectedBucketOwner
<?xml version="1.0" encoding="UTF-8"?>
<SelectObjectContentRequest xmlns="http://s3.amazonaws.com/doc/2006-03-01/">
  <Expression string</Expression>
  <ExpressionType></ExpressionType>
  <RequestProgress></RequestProgress>
  </RequestProgress>
  <InputSerialization></InputSerialization>
  <OutputSerialization></OutputSerialization>
</SelectObjectContentRequest>

API Version 2006-03-01
URI Request Parameters

The request uses the following URI parameters.

**Bucket (p. 388)**

The S3 bucket.

Required: Yes

**Key (p. 388)**

The object key.

Length Constraints: Minimum length of 1.

Required: Yes

**x-amz-expected-bucket-owner (p. 388)**

The account id of the expected bucket owner. If the bucket is owned by a different account, the request will fail with an HTTP 403 (Access Denied) error.

**x-amz-server-side-encryption-customer-algorithm (p. 388)**

The SSE Algorithm used to encrypt the object. For more information, see Server-Side Encryption (Using Customer-Provided Encryption Keys).

**x-amz-server-side-encryption-customer-key (p. 388)**

The SSE Customer Key. For more information, see Server-Side Encryption (Using Customer-Provided Encryption Keys).

**x-amz-server-side-encryption-customer-key-MD5 (p. 388)**

The SSE Customer Key MD5. For more information, see Server-Side Encryption (Using Customer-Provided Encryption Keys).

Request Body

The request accepts the following data in XML format.

**SelectObjectContentRequest (p. 388)**

Root level tag for the SelectObjectContentRequest parameters.

Required: Yes

**Expression (p. 388)**

The expression that is used to query the object.

Type: String

Required: Yes

**ExpressionType (p. 388)**

The type of the provided expression (for example, SQL).
Type: String
Valid Values: SQL
Required: Yes

**InputSerialization (p. 388)**

Describes the format of the data in the object that is being queried.

Type: InputSerialization (p. 571) data type
Required: Yes

**OutputSerialization (p. 388)**

Describes the format of the data that you want Amazon S3 to return in response.

Type: OutputSerialization (p. 616) data type
Required: Yes

**RequestProgress (p. 388)**

Specifies if periodic request progress information should be enabled.

Type: RequestProgress (p. 643) data type
Required: No

**ScanRange (p. 388)**

Specifies the byte range of the object to get the records from. A record is processed when its first byte is contained by the range. This parameter is optional, but when specified, it must not be empty. See RFC 2616, Section 14.35.1 about how to specify the start and end of the range.

ScanRangemay be used in the following ways:

- `<scanrange><start>50</start><end>100</end></scanrange>` - process only the records starting between the bytes 50 and 100 (inclusive, counting from zero)
- `<scanrange><start>50</start></scanrange>` - process only the records starting after the byte 50
- `<scanrange><end>50</end></scanrange>` - process only the records within the last 50 bytes of the file.

Type: ScanRange (p. 652) data type
Required: No

**Response Syntax**

```
HTTP/1.1 200
<?xml version="1.0" encoding="UTF-8"?>
<Payload>
<Records>
PAYLOADblob</Payload>
</Records>
<Stats>
<Details>
<BytesProcessed>long</BytesProcessed>
<BytesReturned>long</BytesReturned>
<BytesScanned>long</BytesScanned>
```
Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in XML format by the service.

Payload (p. 390)

Root level tag for the Payload parameters.

Required: Yes

Cont (p. 390)

The Continuation Event.

Type: ContinuationEvent (p. 532) data type

End (p. 390)

The End Event.

Type: EndEvent (p. 551) data type

Progress (p. 390)

The Progress Event.

Type: ProgressEvent (p. 624) data type

Records (p. 390)

The Records Event.

Type: RecordsEvent (p. 631) data type

Stats (p. 390)

The Stats Event.

Type: StatsEvent (p. 663) data type

Examples

Example 1: CSV object

The following select request retrieves all records from an object with data stored in CSV format. The OutputSerialization element directs Amazon S3 to return results in CSV.
You can try different queries in the Expression element:

- Assuming that you are not using column headers, you can identify columns using positional headers:

  ```sql
  SELECT s._1, s._2 FROM S3Object s WHERE s._3 > 100
  ```

- If you have column headers and you set the FileHeaderInfo to Use, you can identify columns by name in the expression:

  ```sql
  role="nocopy" language="sql">SELECT s.Id, s.FirstName, s.SSN FROM S3Object s
  ```

- You can specify functions in the SQL expression:

  ```sql
  SELECT count(*) FROM S3Object s WHERE s._1 < 1
  ```

Example

The following is a sample response.

```
HTTP/1.1 200 OK
x-amz-id-2: GFihv3y6+kE7KGI1GEkQhU7/2/chR3Yb2fCb2S04nxI423Dqw2XiQ0B/UZ1sYQvPiBLZNRCovw=
x-amz-request-id: 9F341CD3C4BA79E0
Date: Tue, 17 Oct 2017 23:54:05 GMT

A series of messages
```
Example 2: JSON object

The following select request retrieves all records from an object with data stored in JSON format. The OutputSerialization directs Amazon S3 to return results in CSV.

You can try different queries in the Expression element:

- You can filter by string comparison using record keys:
  ```
  SELECT s.country, s.city from S3Object s where s.city = 'Seattle'
  ```
- You can specify functions in the SQL expression:
  ```
  SELECT count(*) FROM S3Object s
  ```

```xml
<SelectRequest>
  <Expression>*/ from S3Object</Expression>
  <ExpressionType>SQL</ExpressionType>
  <InputSerialization>
    <CompressionType>GZIP</CompressionType>
    <JSON>
      <Type>DOCUMENT</Type>
    </JSON>
  </InputSerialization>
  <OutputSerialization>
    <CSV>
      <QuoteFields>ASNEEDED</QuoteFields>
      <RecordDelimiter>
      </RecordDelimiter>
      <FieldDelimiter>,</FieldDelimiter>
      <QuoteCharacter>"</QuoteCharacter>
      <QuoteEscapeCharacter>"</QuoteEscapeCharacter>
    </CSV>
  </OutputSerialization>
</SelectRequest>
```

Example

The following is a sample response.

```
HTTP/1.1 200 OK
x-amz-id-2: GFihv3y6+kE7KG11GEkQhU7/2/chr3Yb2fCb2S04nx1423Dqwg2XiQ0B/U21lzYqvPilZNRcovw=
Date: Tue, 17 Oct 2017 23:54:05 GMT
A series of messages
```

Example 3: Parquet object

- The InputSerialization element describes the format of the data in the object that is being queried. It must specify CSV, JSON, or Parquet.
• The **OutputSerialization** element describes the format of the data that you want Amazon S3 to return in response to the query. It must specify **CSV**, **JSON**. Amazon S3 doesn't support outputting data in the Parquet format.

• The format of the **InputSerialization** doesn't need to match the format of the **OutputSerialization**. So, for example, you can specify **JSON** in the **InputSerialization** and **CSV** in the **OutputSerialization**.

```xml
POST /exampleobject.parquet?select&select-type=2 HTTP/1.1
Host: examplebucket.s3.<Region>.amazonaws.com
Date: Tue, 17 Oct 2017 01:49:52 GMT
Authorization: authorization string
Content-Length: content length

<?xml version="1.0" encoding="UTF-8"?>
<SelectRequest>
  <Expression>Select * from S3Object</Expression>
  <ExpressionType>SQL</ExpressionType>
  <InputSerialization>
    <CompressionType>NONE</CompressionType>
    <Parquet/>
  </InputSerialization>
  <OutputSerialization>
    <CSV>
      <QuoteFields>ASNEEDED</QuoteFields>
      <RecordDelimiter>
      </RecordDelimiter>
      <FieldDelimiter>,</FieldDelimiter>
      <QuoteCharacter>"</QuoteCharacter>
      <QuoteEscapeCharacter>"</QuoteEscapeCharacter>
    </CSV>
  </Output.Serialization>
</SelectRequest>
```

**See Also**

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3
UploadPart

Service: Amazon Simple Storage Service

Uploads a part in a multipart upload.

**Note**

In this operation, you provide part data in your request. However, you have an option to specify your existing Amazon S3 object as a data source for the part you are uploading. To upload a part from an existing object, you use the UploadPartCopy operation.

You must initiate a multipart upload (see CreateMultipartUpload) before you can upload any part. In response to your initiate request, Amazon S3 returns an upload ID, a unique identifier, that you must include in your upload part request.

Part numbers can be any number from 1 to 10,000, inclusive. A part number uniquely identifies a part and also defines its position within the object being created. If you upload a new part using the same part number that was used with a previous part, the previously uploaded part is overwritten. Each part must be at least 5 MB in size, except the last part. There is no size limit on the last part of your multipart upload.

To ensure that data is not corrupted when traversing the network, specify the Content-MD5 header in the upload part request. Amazon S3 checks the part data against the provided MD5 value. If they do not match, Amazon S3 returns an error.

If the upload request is signed with Signature Version 4, then AWS S3 uses the x-amz-content-sha256 header as a checksum instead of Content-MD5. For more information see Authenticating Requests: Using the Authorization Header (AWS Signature Version 4).

**Note:** After you initiate multipart upload and upload one or more parts, you must either complete or abort multipart upload in order to stop getting charged for storage of the uploaded parts. Only after you either complete or abort multipart upload, Amazon S3 frees up the parts storage and stops charging you for the parts storage.

For more information on multipart uploads, go to Multipart Upload Overview in the Amazon Simple Storage Service Developer Guide.

For information on the permissions required to use the multipart upload API, go to Multipart Upload API and Permissions in the Amazon Simple Storage Service Developer Guide.

You can optionally request server-side encryption where Amazon S3 encrypts your data as it writes it to disks in its data centers and decrypts it for you when you access it. You have the option of providing your own encryption key, or you can use the AWS managed encryption keys. If you choose to provide your own encryption key, the request headers you provide in the request must match the headers you used in the request to initiate the upload by using CreateMultipartUpload. For more information, go to Using Server-Side Encryption in the Amazon Simple Storage Service Developer Guide.

Server-side encryption is supported by the S3 Multipart Upload actions. Unless you are using a customer-provided encryption key, you don't need to specify the encryption parameters in each UploadPart request. Instead, you only need to specify the server-side encryption parameters in the initial Initiate Multipart request. For more information, see CreateMultipartUpload.

If you requested server-side encryption using a customer-provided encryption key in your initiate multipart upload request, you must provide identical encryption information in each part upload using the following headers.

- x-amz-server-side-encryption-customer-algorithm
- x-amz-server-side-encryption-customer-key
- `x-amz-server-side-encryption-customer-key-MD5`

**Special Errors**

- **Code:** NoSuchUpload
  - **Cause:** The specified multipart upload does not exist. The upload ID might be invalid, or the multipart upload might have been aborted or completed.
  - **HTTP Status Code:** 404 Not Found
  - **SOAP Fault Code Prefix:** Client

**Related Resources**

- CreateMultipartUpload
- CompleteMultipartUpload
- AbortMultipartUpload
- ListParts
- ListMultipartUploads

**Request Syntax**

```
PUT /Key+?partNumber=PartNumber&uploadId=UploadId HTTP/1.1
Host: Bucket.s3.amazonaws.com
Content-Length: ContentLength
Content-MD5: ContentMD5
x-amz-server-side-encryption-customer-algorithm: SSECustomerAlgorithm
x-amz-server-side-encryption-customer-key: SSECustomerKey
x-amz-server-side-encryption-customer-key-MD5: SSECustomerKeyMD5
x-amz-request-payer: RequestPayer
x-amz-expected-bucket-owner: ExpectedBucketOwner

Body
```

**URI Request Parameters**

The request uses the following URI parameters.

**Bucket (p. 396)**

The name of the bucket to which the multipart upload was initiated.

When using this API with an access point, you must direct requests to the access point hostname. The access point hostname takes the form `AccessPointName-AccountId.s3-accesspoint.Region.amazonaws.com`. When using this operation with an access point through the AWS SDKs, you provide the access point ARN in place of the bucket name. For more information about access point ARNs, see Using Access Points in the Amazon Simple Storage Service Developer Guide.

When using this API with Amazon S3 on Outposts, you must direct requests to the S3 on Outposts hostname. The S3 on Outposts hostname takes the form `AccessPointName-AccountId.outpostId.s3-outposts.Region.amazonaws.com`. When using this operation using S3 on Outposts through the AWS SDKs, you provide the Outposts bucket ARN in place of the bucket name. For more information about S3 on Outposts ARNs, see Using S3 on Outposts in the Amazon Simple Storage Service Developer Guide.

Required: Yes
Content-Length (p. 396)

Size of the body in bytes. This parameter is useful when the size of the body cannot be determined automatically.

Content-MD5 (p. 396)

The base64-encoded 128-bit MD5 digest of the part data. This parameter is auto-populated when using the command from the CLI. This parameter is required if object lock parameters are specified.

Key (p. 396)

Object key for which the multipart upload was initiated.

Length Constraints: Minimum length of 1.

Required: Yes

partNumber (p. 396)

Part number of part being uploaded. This is a positive integer between 1 and 10,000.

Required: Yes

uploadID (p. 396)

Upload ID identifying the multipart upload whose part is being uploaded.

Required: Yes

x-amz-expected-bucket-owner (p. 396)

The account id of the expected bucket owner. If the bucket is owned by a different account, the request will fail with an HTTP 403 (Access Denied) error.

x-amz-request-payer (p. 396)

Confirms that the requester knows that they will be charged for the request. Bucket owners need not specify this parameter in their requests. For information about downloading objects from requester pays buckets, see Downloading Objects in Requestor Pays Buckets in the Amazon S3 Developer Guide.

Valid Values: requester

x-amz-server-side-encryption-customer-algorithm (p. 396)

Specifies the algorithm to use to when encrypting the object (for example, AES256).

x-amz-server-side-encryption-customer-key (p. 396)

Specifies the customer-provided encryption key for Amazon S3 to use in encrypting data. This value is used to store the object and then it is discarded; Amazon S3 does not store the encryption key. The key must be appropriate for use with the algorithm specified in the x-amz-server-side-encryption-customer-algorithm header. This must be the same encryption key specified in the initiate multipart upload request.

x-amz-server-side-encryption-customer-key-MD5 (p. 396)

Specifies the 128-bit MD5 digest of the encryption key according to RFC 1321. Amazon S3 uses this header for a message integrity check to ensure that the encryption key was transmitted without error.

Request Body

The request accepts the following binary data.
Response Syntax

HTTP/1.1 200
x-amz-server-side-encryption: ServerSideEncryption
ETag: ETag
x-amz-server-side-encryption-customer-algorithm: SSECustomerAlgorithm
x-amz-server-side-encryption-customer-key-MD5: SSECustomerKeyMD5
x-amz-server-side-encryption-aws-kms-key-id: SSEKMSKeyId
x-amz-request-charged: RequestCharged

Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The response returns the following HTTP headers.

**ETag (p. 398)**

Entity tag for the uploaded object.

**x-amz-request-charged (p. 398)**

If present, indicates that the requester was successfully charged for the request.

Valid Values: requester

**x-amz-server-side-encryption (p. 398)**

The server-side encryption algorithm used when storing this object in Amazon S3 (for example, AES256, aws:kms).

Valid Values: AES256 | aws:kms

**x-amz-server-side-encryption-aws-kms-key-id (p. 398)**

If present, specifies the ID of the AWS Key Management Service (AWS KMS) symmetric customer managed customer master key (CMK) was used for the object.

**x-amz-server-side-encryption-customer-algorithm (p. 398)**

If server-side encryption with a customer-provided encryption key was requested, the response will include this header confirming the encryption algorithm used.

**x-amz-server-side-encryption-customer-key-MD5 (p. 398)**

If server-side encryption with a customer-provided encryption key was requested, the response will include this header to provide round-trip message integrity verification of the customer-provided encryption key.

Examples

Sample Request

The following PUT request uploads a part (part number 1) in a multipart upload. The request includes the upload ID that you get in response to your Initiate Multipart Upload request.

```
PUT /my-movie.m2ts?
partNumber=1&uploadId=VCVsb2FkIElEIGZvciBlbZZpbmcncyBteS1tb3ZpZS5tMnRzIHVwbG9hZR HTTP/1.1
Host: example-bucket.s3.<Region>.amazonaws.com
```
Sample Response

The response includes the ETag header. You need to retain this value for use when you send the Complete Multipart Upload request.

Example: Upload a part with an encryption key in the request for server-side encryption

If you initiated a multipart upload with a request to save an object using server-side encryption with a customer-provided encryption key, each part upload must also include the same set of encryption-specific headers as shown in the following example request.

Example

In the response, Amazon S3 returns encryption-specific headers providing the encryption algorithm used and MD5 digest of the encryption key you provided in the request.

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:
• AWS Command Line Interface
• AWS SDK for .NET
• AWS SDK for C++
• AWS SDK for Go
• AWS SDK for Java
• AWS SDK for JavaScript
• AWS SDK for PHP V3
• AWS SDK for Python
• AWS SDK for Ruby V3
UploadPartCopy
Service: Amazon Simple Storage Service

Uploads a part by copying data from an existing object as data source. You specify the data source by adding the request header `x-amz-copy-source` in your request and a byte range by adding the request header `x-amz-copy-source-range` in your request.

The minimum allowable part size for a multipart upload is 5 MB. For more information about multipart upload limits, go to Quick Facts in the Amazon Simple Storage Service Developer Guide.

**Note**

Instead of using an existing object as part data, you might use the UploadPart operation and provide data in your request.

You must initiate a multipart upload before you can upload any part. In response to your initiate request, Amazon S3 returns a unique identifier, the upload ID, that you must include in your upload part request.

For more information about using the UploadPartCopy operation, see the following:

- For conceptual information about multipart uploads, see Uploading Objects Using Multipart Upload in the Amazon Simple Storage Service Developer Guide.
- For information about permissions required to use the multipart upload API, see Multipart Upload API and Permissions in the Amazon Simple Storage Service Developer Guide.
- For information about copying objects using a single atomic operation vs. the multipart upload, see Operations on Objects in the Amazon Simple Storage Service Developer Guide.
- For information about using server-side encryption with customer-provided encryption keys with the UploadPartCopy operation, see CopyObject and UploadPart.

Note the following additional considerations about the request headers `x-amz-copy-source-if-match`, `x-amz-copy-source-if-none-match`, `x-amz-copy-source-if-unmodified-since`, and `x-amz-copy-source-if-modified-since`:

**Consideration 1** - If both of the `x-amz-copy-source-if-match` and `x-amz-copy-source-if-unmodified-since` headers are present in the request as follows:

- `x-amz-copy-source-if-match` condition evaluates to true, and;
- `x-amz-copy-source-if-unmodified-since` condition evaluates to false;

Amazon S3 returns 200 OK and copies the data.

**Consideration 2** - If both of the `x-amz-copy-source-if-none-match` and `x-amz-copy-source-if-modified-since` headers are present in the request as follows:

- `x-amz-copy-source-if-none-match` condition evaluates to false, and;
- `x-amz-copy-source-if-modified-since` condition evaluates to true;

Amazon S3 returns 412 Precondition Failed response code.

**Versioning**

If your bucket has versioning enabled, you could have multiple versions of the same object. By default, `x-amz-copy-source` identifies the current version of the object to copy. If the current version is a delete marker and you don't specify a versionId in the `x-amz-copy-source`, Amazon S3 returns a 404 error, because the object does not exist. If you specify versionId in the `x-amz-copy-source` and the versionId is a delete marker, Amazon S3 returns an HTTP 400 error, because you are not allowed to specify a delete marker as a version for the `x-amz-copy-source`. 
You can optionally specify a specific version of the source object to copy by adding the `versionId` subresource as shown in the following example:

```
x-amz-copy-source: /bucket/object?versionId=version id
```

**Special Errors**

- **Code: NoSuchUpload**
  - **Cause:** The specified multipart upload does not exist. The upload ID might be invalid, or the multipart upload might have been aborted or completed.
  - **HTTP Status Code:** 404 Not Found
- **Code: InvalidRequest**
  - **Cause:** The specified copy source is not supported as a byte-range copy source.
  - **HTTP Status Code:** 400 Bad Request

**Related Resources**

- CreateMultipartUpload
- UploadPart
- CompleteMultipartUpload
- AbortMultipartUpload
- ListParts
- ListMultipartUploads

**Request Syntax**

```
PUT /Key+?partNumber=PartNumber&uploadId=UploadId HTTP/1.1
Host: Bucket.s3.amazonaws.com
x-amz-copy-source: CopySource
x-amz-copy-source-if-match: CopySourceIfMatch
x-amz-copy-source-if-modified-since: CopySourceIfModifiedSince
x-amz-copy-source-if-none-match: CopySourceIfNoneMatch
x-amz-copy-source-if-unmodified-since: CopySourceIfUnmodifiedSince
x-amz-copy-source-range: CopySourceRange
x-amz-server-side-encryption-customer-algorithm: SSECustomerAlgorithm
x-amz-server-side-encryption-customer-key: SSECustomerKey
x-amz-server-side-encryption-customer-key-MD5: SSECustomerKeyMD5
x-amz-copy-source-server-side-encryption-customer-algorithm: CopySourceSSECustomerAlgorithm
x-amz-copy-source-server-side-encryption-customer-key: CopySourceSSECustomerKey
x-amz-copy-source-server-side-encryption-customer-key-MD5: CopySourceSSECustomerKeyMD5
x-amz-request-payer: RequestPayer
x-amz-expected-bucket-owner: ExpectedBucketOwner
x-amz-source-expected-bucket-owner: ExpectedSourceBucketOwner
```

**URI Request Parameters**

The request uses the following URI parameters.

**Bucket (p. 402)**

The bucket name.

When using this API with an access point, you must direct requests to the access point hostname. The access point hostname takes the form `AccessPointName-AccountId.s3-accesspoint.Region.amazonaws.com`. When using this operation with an access point through the
AWS SDKs, you provide the access point ARN in place of the bucket name. For more information about access point ARNs, see Using Access Points in the Amazon Simple Storage Service Developer Guide.

When using this API with Amazon S3 on Outposts, you must direct requests to the S3 on Outposts hostname. The S3 on Outposts hostname takes the form AccessPointName-AccountId.outpostId.s3-outposts.Region.amazonaws.com. When using this operation using S3 on Outposts through the AWS SDKs, you provide the Outposts bucket ARN in place of the bucket name. For more information about S3 on Outposts ARNs, see Using S3 on Outposts in the Amazon Simple Storage Service Developer Guide.

Required: Yes

Key (p. 402)

Object key for which the multipart upload was initiated.

Length Constraints: Minimum length of 1.

Required: Yes

partNumber (p. 402)

Part number of part being copied. This is a positive integer between 1 and 10,000.

Required: Yes

uploadId (p. 402)

Upload ID identifying the multipart upload whose part is being copied.

Required: Yes

x-amz-copy-source (p. 402)

Specifies the source object for the copy operation. You specify the value in one of two formats, depending on whether you want to access the source object through an access point:

- For objects not accessed through an access point, specify the name of the source bucket and key of the source object, separated by a slash (/). For example, to copy the object reports/january.pdf from the bucket awsexamplebucket, use awsexamplebucket/reports/january.pdf. The value must be URL encoded.

- For objects accessed through access points, specify the Amazon Resource Name (ARN) of the object as accessed through the access point, in the format arn:aws:s3:<Region>:<account-id>:accesspoint/<access-point-name>/object/<key>. For example, to copy the object reports/january.pdf through access point my-access-point owned by account 123456789012 in Region us-west-2, use the URL encoding of arn:aws:s3:us-west-2:123456789012:accesspoint/my-access-point/object/reports/january.pdf. The value must be URL encoded.

Note

Amazon S3 supports copy operations using access points only when the source and destination buckets are in the same AWS Region.

Alternatively, for objects accessed through Amazon S3 on Outposts, specify the ARN of the object as accessed through the format arn:aws:s3-outposts:<Region>:<account-id>:outpost/<outpost-id>/object/<key>. For example, to copy the object reports/january.pdf through outpost my-outpost owned by account 123456789012 in Region us-west-2, use the URL encoding of arn:aws:s3-outposts:us-west-2:123456789012:outpost/my-outpost/object/reports/january.pdf. The value must be URL encoded.

To copy a specific version of an object, append ?versionId=<version-id> to the value (for example, awsexamplebucket/reports/january.pdf?
versionId=QUpfdndhfd8438MNFDN93jdJFkdmqnh893). If you don't specify a version ID, Amazon S3 copies the latest version of the source object.

Pattern: \\/.+\\/.+

Required: Yes

**x-amz-copy-source-if-match** (p. 402)

Copies the object if its entity tag (ETag) matches the specified tag.

**x-amz-copy-source-if-modified-since** (p. 402)

Copies the object if it has been modified since the specified time.

**x-amz-copy-source-if-none-match** (p. 402)

Copies the object if its entity tag (ETag) is different than the specified ETag.

**x-amz-copy-source-if-unmodified-since** (p. 402)

Copies the object if it hasn't been modified since the specified time.

**x-amz-copy-source-range** (p. 402)

The range of bytes to copy from the source object. The range value must use the form bytes=first-last, where the first and last are the zero-based byte offsets to copy. For example, bytes=0-9 indicates that you want to copy the first 10 bytes of the source. You can copy a range only if the source object is greater than 5 MB.

**x-amz-copy-source-server-side-encryption-customer-algorithm** (p. 402)

Specifies the algorithm to use when decrypting the source object (for example, AES256).

**x-amz-copy-source-server-side-encryption-customer-key** (p. 402)

Specifies the customer-provided encryption key for Amazon S3 to use to decrypt the source object. The encryption key provided in this header must be one that was used when the source object was created.

**x-amz-copy-source-server-side-encryption-customer-key-MD5** (p. 402)

Specifies the 128-bit MD5 digest of the encryption key according to RFC 1321. Amazon S3 uses this header for a message integrity check to ensure that the encryption key was transmitted without error.

**x-amz-expected-bucket-owner** (p. 402)

The account id of the expected destination bucket owner. If the destination bucket is owned by a different account, the request will fail with an HTTP 403 (Access Denied) error.

**x-amz-request-payer** (p. 402)

Confirms that the requester knows that they will be charged for the request. Bucket owners need not specify this parameter in their requests. For information about downloading objects from requester pays buckets, see Downloading Objects in Requestor Pays Buckets in the Amazon S3 Developer Guide.

Valid Values: requester

**x-amz-server-side-encryption-customer-algorithm** (p. 402)

Specifies the algorithm to use to when encrypting the object (for example, AES256).

**x-amz-server-side-encryption-customer-key** (p. 402)

Specifies the customer-provided encryption key for Amazon S3 to use in encrypting data. This value is used to store the object and then it is discarded; Amazon S3 does not store the encryption key.
The key must be appropriate for use with the algorithm specified in the `x-amz-server-side-encryption-customer-algorithm` header. This must be the same encryption key specified in the initiate multipart upload request.

**x-amz-server-side-encryption-customer-key-MD5 (p. 402)**

Specifies the 128-bit MD5 digest of the encryption key according to RFC 1321. Amazon S3 uses this header for a message integrity check to ensure that the encryption key was transmitted without error.

**x-amz-source-expected-bucket-owner (p. 402)**

The account ID of the expected source bucket owner. If the source bucket is owned by a different account, the request will fail with an HTTP 403 (Access Denied) error.

**Request Body**

The request does not have a request body.

**Response Syntax**

```
HTTP/1.1 200
x-amz-copy-source-version-id: CopySourceVersionId
x-amz-server-side-encryption: ServerSideEncryption
x-amz-server-side-encryption-customer-algorithm: SSECustomerAlgorithm
x-amz-server-side-encryption-customer-key-MD5: SSECustomerKeyMD5
x-amz-server-side-encryption-aws-kms-key-id: SSEKMSKeyId
x-amz-request-charged: RequestCharged
<?xml version="1.0" encoding="UTF-8"?>
<CopyPartResult>
  <ETag>string</ETag>
  <LastModified>timestamp</LastModified>
</CopyPartResult>
```

**Response Elements**

If the action is successful, the service sends back an HTTP 200 response.

The response returns the following HTTP headers.

**x-amz-copy-source-version-id (p. 405)**

The version of the source object that was copied, if you have enabled versioning on the source bucket.

**x-amz-request-charged (p. 405)**

If present, indicates that the requester was successfully charged for the request.

Valid Values: requester

**x-amz-server-side-encryption (p. 405)**

The server-side encryption algorithm used when storing this object in Amazon S3 (for example, AES256, aws:kms).

Valid Values: AES256 | aws:kms

**x-amz-server-side-encryption-aws-kms-key-id (p. 405)**

If present, specifies the ID of the AWS Key Management Service (AWS KMS) symmetric customer managed customer master key (CMK) that was used for the object.
x-amz-server-side-encryption-customer-algorithm (p. 405)
If server-side encryption with a customer-provided encryption key was requested, the response will include this header confirming the encryption algorithm used.

x-amz-server-side-encryption-customer-key-MD5 (p. 405)
If server-side encryption with a customer-provided encryption key was requested, the response will include this header to provide round-trip message integrity verification of the customer-provided encryption key.

The following data is returned in XML format by the service.

CopyPartResult (p. 405)
Root level tag for the CopyPartResult parameters.

Required: Yes

ETag (p. 405)
Entity tag of the object.

Type: String

LastModified (p. 405)
Date and time at which the object was uploaded.

Type: Timestamp

Examples

Sample Request
The following PUT request uploads a part (part number 2) in a multipart upload. The request specifies a byte range from an existing object as the source of this upload. The request includes the upload ID that you get in response to your Initiate Multipart Upload request.

```
PUT /newobject?
partNumber=2&uploadId=VCVsb2FkIElEIGZvciBteS1tb3ZpZmcncyBteS1tb3ZpZS5tMnRzIHVwbG9hZR HTTP/1.1
Host: target-bucket.s3.<Region>.amazonaws.com
Date: Mon, 11 Apr 2011 20:34:56 GMT
x-amz-copy-source: /source-bucket/sourceobject
x-amz-copy-source-range:bytes=500-6291456
Authorization: authorization string
```

Sample Response
The response includes the ETag value. You need to retain this value to use when you send the Complete Multipart Upload request.

```
HTTP/1.1 200 OK
x-amz-id-2: Vvag1LuByRx9e6j5Onimru9p04ZVKnJ2Qz7/C1NtcfTWatRPfTaOFg==
x-amz-request-id: 656c76696e6727732072657175657374
Date: Mon, 11 Apr 2011 20:34:56 GMT
Server: AmazonS3
```
Sample Request

The following PUT request uploads a part (part number 2) in a multipart upload. The request does not specify the optional byte range header, but requests the entire source object copy as part 2. The request includes the upload ID that you got in response to your Initiate Multipart Upload request.

PUT /newobject?partNumber=2&uploadId=VCVsb2FkIElEIGZvciBlbZZpbmcncyBteS1tb3ZpZS5tMnRzIHVwbG9hZR HTTP/1.1
Host: target-bucket.s3.<Region>.amazonaws.com
Date: Mon, 11 Apr 2011 20:34:56 GMT
x-amz-copy-source: /source-bucket/sourceobject?versionId=3/L4kqtJlcpXroDTDmJ+rmSpXd3dbrH
x-amz-copy-source-range: bytes=500-6291456
Authorization: authorization string

Sample Response

The response includes the ETag value. You need to retain this value to use when you send the Complete Multipart Upload request.

HTTP/1.1 200 OK
x-amz-id-2: Vvag1LuByRx9e6j5Onimru9pO4ZVKnJ3Qz7/C1NPcfTWATpFtaOFg==
x-amz-request-id: 656c76696e6727732072657175657374
x-amz-copy-source-version-id: 3/L4kqtJlcpXroDTDmJ+rmSpXd3dbrH
x-amz-copy-source-range: bytes=500-6291456
Date: Mon, 11 Apr 2011 20:34:56 GMT
Server: AmazonS3

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3
AWS S3 Control

The following actions are supported by AWS S3 Control:

- CreateAccessPoint (p. 409)
- CreateBucket (p. 413)
- CreateJob (p. 417)
- DeleteAccessPoint (p. 422)
- DeleteAccessPointPolicy (p. 424)
- DeleteBucket (p. 426)
- DeleteBucketLifecycleConfiguration (p. 428)
- DeleteBucketPolicy (p. 430)
- DeleteBucketTagging (p. 432)
- DeleteJobTagging (p. 434)
- DeletePublicAccessBlock (p. 436)
- DescribeJob (p. 438)
- GetAccessPoint (p. 442)
- GetAccessPointPolicy (p. 445)
- GetAccessPointPolicyStatus (p. 447)
- GetBucket (p. 449)
- GetBucketLifecycleConfiguration (p. 452)
- GetBucketPolicy (p. 456)
- GetBucketTagging (p. 459)
- GetJobTagging (p. 462)
- GetPublicAccessBlock (p. 464)
- ListAccessPoints (p. 467)
- ListJobs (p. 470)
- ListRegionalBuckets (p. 473)
- PutAccessPointPolicy (p. 476)
- PutBucketLifecycleConfiguration (p. 479)
- PutBucketPolicy (p. 482)
- PutBucketTagging (p. 485)
- PutJobTagging (p. 488)
- PutPublicAccessBlock (p. 491)
- UpdateJobPriority (p. 494)
- UpdateJobStatus (p. 496)
CreateAccessPoint
Service: AWS S3 Control

Creates an access point and associates it with the specified bucket. For more information, see Managing Data Access with Amazon S3 Access Points in the Amazon Simple Storage Service Developer Guide.

Using this action with Amazon S3 on Outposts

This action:

- Requires a virtual private cloud (VPC) configuration as S3 on Outposts only supports VPC style access points.
- Does not support ACL on S3 on Outposts buckets.
- Does not support Public Access on S3 on Outposts buckets.
- Does not support object lock for S3 on Outposts buckets.

For more information, see Using Amazon S3 on Outposts in the Amazon Simple Storage Service Developer Guide.

All Amazon S3 on Outposts REST API requests for this action require an additional parameter of x-amz-outpost-id to be passed with the request and an S3 on Outposts endpoint hostname prefix instead of s3-control. For an example of the request syntax for Amazon S3 on Outposts that uses the S3 on Outposts endpoint hostname prefix and the x-amz-outpost-id derived using the access point ARN, see the Examples section.

The following actions are related to CreateAccessPoint:

- GetAccessPoint
- DeleteAccessPoint
- ListAccessPoints

Request Syntax

```
PUT /v20180820/accesspoint/name HTTP/1.1
x-amz-account-id: AccountId
<?xml version="1.0" encoding="UTF-8"?>
<CreateAccessPointRequest xmlns="http://awss3control.amazonaws.com/doc/2018-08-20/">
  <Bucket>string</Bucket>
  <VpcConfiguration>
    <VpcId>string</VpcId>
  </VpcConfiguration>
  <PublicAccessBlockConfiguration>
    <BlockPublicAcls>boolean</BlockPublicAcls>
    <BlockPublicPolicy>boolean</BlockPublicPolicy>
    <IgnorePublicAcls>boolean</IgnorePublicAcls>
    <RestrictPublicBuckets>boolean</RestrictPublicBuckets>
  </PublicAccessBlockConfiguration>
</CreateAccessPointRequest>
```

URI Request Parameters

The request uses the following URI parameters.

name (p. 409)

The name you want to assign to this access point.

Required: Yes

x-amz-account-id (p. 409)

The AWS account ID for the owner of the bucket for which you want to create an access point.

Length Constraints: Maximum length of 64.

Pattern: ^\d{12}$

Required: Yes

Request Body

The request accepts the following data in XML format.

CreateAccessPointRequest (p. 409)

Root level tag for the CreateAccessPointRequest parameters.

Required: Yes

Bucket (p. 409)

The name of the bucket that you want to associate this access point with.

For using this parameter with Amazon S3 on Outposts with the REST API, you must specify the name and the x-amz-outpost-id as well.

For using this parameter with S3 on Outposts with the AWS SDK and CLI, you must specify the ARN of the bucket accessed in the format arn:aws:s3-outposts:<Region>:<account-id>:outpost/<outpost-id>/bucket/<my-bucket-name>. For example, to access the bucket reports through outpost my-outpost owned by account 123456789012 in Region us-west-2, use the URL encoding of arn:aws:s3-outposts:us-west-2:123456789012:outpost/my-outpost/bucket/reports. The value must be URL encoded.

Type: String


Required: Yes

PublicAccessBlockConfiguration (p. 409)

The PublicAccessBlock configuration that you want to apply to this Amazon S3 account. You can enable the configuration options in any combination. For more information about when Amazon S3 considers a bucket or object public, see The Meaning of “Public” in the Amazon Simple Storage Service Developer Guide.

This is not supported for Amazon S3 on Outposts.

Type: PublicAccessBlockConfiguration (p. 706) data type

Required: No

VpcConfiguration (p. 409)

If you include this field, Amazon S3 restricts access to this access point to requests from the specified virtual private cloud (VPC).  

Note
This is required for creating an access point for Amazon S3 on Outposts buckets.
Type: VpcConfiguration (p. 729) data type

Required: No

Response Syntax

HTTP/1.1 200
<?xml version="1.0" encoding="UTF-8"?>
<CreateAccessPointResult>
<AccessPointArn>string</AccessPointArn>
</CreateAccessPointResult>

Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in XML format by the service.

CreateAccessPointResult (p. 411)

Root level tag for the CreateAccessPointResult parameters.

Required: Yes

AccessPointArn (p. 411)

The ARN of the access point.

Note
This is only supported by Amazon S3 on Outposts.

Type: String


Examples

Sample request for creating an access point for an Amazon S3 on Outposts bucket

This request creates an access point for S3 on Outposts bucket.

PUT /v20180820/accesspoint/example-access-point HTTP/1.1
Host:s3-outposts.<Region>.amazonaws.com
x-amz-account-id: example-account-id
x-amz-outpost-id: op-01ac5d28a6a232904
<?xml version="1.0" encoding="UTF-8"?>
<CreateAccessPointRequest xmlns="http://awss3control.amazonaws.com/doc/2018-08-20/"
<Bucket>example-outpost.bucket</Bucket>
</CreateAccessPointRequest>

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
• AWS SDK for C++
• AWS SDK for Go
• AWS SDK for Java
• AWS SDK for JavaScript
• AWS SDK for PHP V3
• AWS SDK for Python
• AWS SDK for Ruby V3
CreateBucket
Service: AWS S3 Control

Note
This API operation creates an Amazon S3 on Outposts bucket. To create an S3 bucket, see Create Bucket in the Amazon Simple Storage Service API.

Creates a new Outposts bucket. By creating the bucket, you become the bucket owner. To create an Outposts bucket, you must have S3 on Outposts. For more information, see Using Amazon S3 on Outposts in Amazon Simple Storage Service Developer Guide.

Not every string is an acceptable bucket name. For information on bucket naming restrictions, see Working with Amazon S3 Buckets.

S3 on Outposts buckets do not support

- ACLs. Instead, configure access point policies to manage access to buckets.
- Public access.
- Object Lock
- Bucket Location constraint

For an example of the request syntax for Amazon S3 on Outposts that uses the S3 on Outposts endpoint hostname prefix and x-amz-outpost-id in your API request, see the Examples section.

The following actions are related to CreateBucket for Amazon S3 on Outposts:

- PutObject
- GetBucket
- DeleteBucket
- CreateAccessPoint
- PutAccessPointPolicy

Request Syntax

```plaintext
PUT /v20180820/bucket/name HTTP/1.1
Host: Bucket.s3-control.amazonaws.com
x-amz-acl: ACL
x-amz-grant-full-control: GrantFullControl
x-amz-grant-read: GrantRead
x-amz-grant-read-acp: GrantReadACP
x-amz-grant-write: GrantWrite
x-amz-grant-write-acp: GrantWriteACP
x-amz-bucket-object-lock-enabled: ObjectLockEnabledForBucket
x-amz-outpost-id: OutpostId
<?xml version="1.0" encoding="UTF-8"?>
<CreateBucketConfiguration xmlns="http://awss3control.amazonaws.com/doc/2018-08-20/">
  <LocationConstraint>string</LocationConstraint>
</CreateBucketConfiguration>
```

URI Request Parameters

The request uses the following URI parameters.

**name (p. 413)**

The name of the bucket.

Required: Yes

**x-amz-acl** *(p. 413)*

The canned ACL to apply to the bucket.

**Note**
This is not supported by Amazon S3 on Outposts buckets.

Valid Values: private | public-read | public-read-write | authenticated-read

**x-amz.bucket-object-lock-enabled** *(p. 413)*

Specifies whether you want S3 Object Lock to be enabled for the new bucket.

**Note**
This is not supported by Amazon S3 on Outposts buckets.

**x-amz-grant-full-control** *(p. 413)*

Allows grantee the read, write, read ACP, and write ACP permissions on the bucket.

**Note**
This is not supported by Amazon S3 on Outposts buckets.

**x-amz-grant-read** *(p. 413)*

Allows grantee to list the objects in the bucket.

**Note**
This is not supported by Amazon S3 on Outposts buckets.

**x-amz-grant-read-acp** *(p. 413)*

Allows grantee to read the bucket ACL.

**Note**
This is not supported by Amazon S3 on Outposts buckets.

**x-amz-grant-write** *(p. 413)*

Allows grantee to create, overwrite, and delete any object in the bucket.

**Note**
This is not supported by Amazon S3 on Outposts buckets.

**x-amz-grant-write-acp** *(p. 413)*

Allows grantee to write the ACL for the applicable bucket.

**Note**
This is not supported by Amazon S3 on Outposts buckets.

**x-amz-outpost-id** *(p. 413)*

The ID of the Outposts where the bucket is being created.

**Note**
This is required by Amazon S3 on Outposts buckets.

Length Constraints: Minimum length of 1. Maximum length of 64.

**Request Body**

The request accepts the following data in XML format.
CreateBucketConfiguration (p. 413)

Root level tag for the CreateBucketConfiguration parameters.

Required: Yes

LocationConstraint (p. 413)

Specifies the Region where the bucket will be created. If you are creating a bucket on the US East (N. Virginia) Region (us-east-1), you do not need to specify the location.

Note
This is not supported by Amazon S3 on Outposts buckets.

Type: String

Valid Values: EU | eu-west-1 | us-west-1 | us-west-2 | ap-south-1 | ap-southeast-1 | ap-southeast-2 | ap-northeast-1 | sa-east-1 | cn-north-1 | eu-central-1

Required: No

Response Syntax

HTTP/1.1 200
Location: Location
<?xml version="1.0" encoding="UTF-8"?>
<CreateBucketResult>
  <BucketArn>string</BucketArn>
</CreateBucketResult>

Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The response returns the following HTTP headers.

Location (p. 415)

The location of the bucket.

The following data is returned in XML format by the service.

CreateBucketResult (p. 415)

Root level tag for the CreateBucketResult parameters.

Required: Yes

BucketArn (p. 415)

The Amazon Resource Name (ARN) of the bucket.

For using this parameter with Amazon S3 on Outposts with the REST API, you must specify the name and the x-amz-outpost-id as well.

For using this parameter with S3 on Outposts with the AWS SDK and CLI, you must specify the ARN of the bucket accessed in the format arn:aws:s3-outposts:<Region>:<account-id>:outpost/<outpost-id>/bucket/<my-bucket-name>. For example, to access the bucket reports through outpost my-outpost owned by account 123456789012 in Region us-west-2,
use the URL encoding of `arn:aws:s3-outposts:us-west-2:123456789012:outpost/my-outpost/bucket/reports`. The value must be URL encoded.

**Type:** String

**Length Constraints:** Minimum length of 4. Maximum length of 128.

The requested Outposts bucket name is not available. The bucket namespace is shared by all users of the AWS Outposts in this Region. Select a different name and try again. The Outposts bucket you tried to create already exists, and you own it.

**Examples**

**Sample request to create an Amazon S3 on Outposts bucket**

This request creates an Outposts bucket named `example-outpost-bucket`.

```
PUT /v20180820/bucket/example-outpost-bucket/  HTTP/1.1
Host:s3-outposts.<Region>.amazonaws.com
x-amz-outpost-id: op-01ac5d28a6a232904
Content-Length: 0
Date: Wed, 01 Mar 2006 12:00:00 GMT
Authorization: authorization string
```

**See Also**

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3
CreateJob
Service: AWS S3 Control

S3 Batch Operations performs large-scale Batch Operations on Amazon S3 objects. Batch Operations can run a single operation or action on lists of Amazon S3 objects that you specify. For more information, see S3 Batch Operations in the Amazon Simple Storage Service Developer Guide.

This operation creates an S3 Batch Operations job.

Related actions include:

- DescribeJob
- ListJobs
- UpdateJobPriority
- UpdateJobStatus

Request Syntax

POST /v20180820/jobs HTTP/1.1
x-amz-account-id: AccountId
<?xml version="1.0" encoding="UTF-8"?>
<CreateJobRequest xmlns="http://awss3control.amazonaws.com/doc/2018-08-20/"/>
<ConfirmationRequired>boolean</ConfirmationRequired>
<Operation>
  <LambdaInvoke>
    <FunctionArn>string</FunctionArn>
  </LambdaInvoke>
  <S3InitiateRestoreObject>
    <ExpirationInDays>integer</ExpirationInDays>
    <GlacierJobTier>string</GlacierJobTier>
  </S3InitiateRestoreObject>
  <S3PutObjectAcl>
    <AccessControlPolicy>
      <AccessControlList>
        <Grants>
          <S3Grant>
            <Grantee>
              <DisplayName>string</DisplayName>
              <Identifier>string</Identifier>
              <TypeIdentifier>string</TypeIdentifier>
            </Grantee>
            <Permission>string</Permission>
          </S3Grant>
        </Grants>
        <Owner>
          <DisplayName>string</DisplayName>
          <ID>string</ID>
        </Owner>
      </AccessControlList>
      <CannedAccessControlList>string</CannedAccessControlList>
    </AccessControlPolicy>
  </S3PutObjectAcl>
  <S3PutObjectCopy>
    <AccessControlGrants>
      <S3Grant>
        <Grantee>
          <DisplayName>string</DisplayName>
          <Identifier>string</Identifier>
          <TypeIdentifier>string</TypeIdentifier>
        </Grantee>
        <Permission>string</Permission>
      </S3Grant>
    </AccessControlGrants>
  </S3PutObjectCopy>
</Operation>
</S3Grant>
</AccessControlGrants>
<CannedAccessControlList>string</CannedAccessControlList>
<MetadataDirective>string</MetadataDirective>
<ModifiedSinceConstraint>timestamp</ModifiedSinceConstraint>

>NewObjectMetadata>

<CacheControl>string</CacheControl>
<ContentDisposition>string</ContentDisposition>
<ContentEncoding>string</ContentEncoding>
<ContentLanguage>string</ContentLanguage>
<ContentLength>long</ContentLength>
<ContentMD5>string</ContentMD5>
<ContentType>string</ContentType>
<HttpExpiresDate>timestamp</HttpExpiresDate>
<RequesterCharged>boolean</RequesterCharged>

</NewObjectMetadata>
</Report>

</Operation>

</S3PutObjectCopy>
</S3PutObjectLegalHold>

</S3PutObjectLegalHold>
</S3PutObjectRetention>

</S3PutObjectRetention>
</S3PutObjectTagging>
</TagSet>
</S3PutObjectTagging>
</Operation>

</S3PutObjectCopy>
</S3PutObjectLegalHold>

</S3PutObjectLegalHold>
</S3PutObjectRetention>

</S3PutObjectRetention>
</S3PutObjectTagging>
</TagSet>
</S3PutObjectTagging>
<ReportScope>string</ReportScope>
</Report>
<ClientRequestToken>string</ClientRequestToken>
<Manifest>
  <Location>
    <ETag>string</ETag>
    <ObjectArn>string</ObjectArn>
    <ObjectVersionId>string</ObjectVersionId>
  </Location>
  <Spec>
    <Fields>
      <INVALID-TYPE-NAME>string</INVALID-TYPE-NAME>
    </Fields>
    <Format>string</Format>
  </Spec>
</Manifest>
<Description>string</Description>
<Priority>integer</Priority>
<RoleArn>string</RoleArn>
<Tag>
  <Key>string</Key>
  <Value>string</Value>
</Tag>
</Tags>
</CreateJobRequest>

**URI Request Parameters**

The request uses the following URI parameters.

**x-amz-account-id (p. 417)**

The AWS account ID that creates the job.

Length Constraints: Maximum length of 64.

Pattern: ^\d{12}$

Required: Yes

**Request Body**

The request accepts the following data in XML format.

**CreateJobRequest (p. 417)**

Root level tag for the CreateJobRequest parameters.

Required: Yes

**ClientRequestToken (p. 417)**

An idempotency token to ensure that you don't accidentally submit the same request twice. You can use any string up to the maximum length.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 64.

Required: Yes
ConfirmationRequired (p. 417)
Indicates whether confirmation is required before Amazon S3 runs the job. Confirmation is only required for jobs created through the Amazon S3 console.

Type: Boolean
Required: No

Description (p. 417)
A description for this job. You can use any string within the permitted length. Descriptions don't need to be unique and can be used for multiple jobs.

Type: String
Length Constraints: Minimum length of 1. Maximum length of 256.
Required: No

Manifest (p. 417)
Configuration parameters for the manifest.

Type: JobManifest (p. 688) data type
Required: Yes

Operation (p. 417)
The operation that you want this job to perform on each object listed in the manifest. For more information about the available operations, see Operations in the Amazon Simple Storage Service Developer Guide.

Type: JobOperation (p. 691) data type
Required: Yes

Priority (p. 417)
The numerical priority for this job. Higher numbers indicate higher priority.

Type: Integer
Valid Range: Minimum value of 0. Maximum value of 2147483647.
Required: Yes

Report (p. 417)
Configuration parameters for the optional job-completion report.

Type: JobReport (p. 694) data type
Required: Yes

RoleArn (p. 417)
The Amazon Resource Name (ARN) for the AWS Identity and Access Management (IAM) role that Batch Operations will use to run this job's operation on each object in the manifest.

Type: String
Length Constraints: Minimum length of 1. Maximum length of 2048.
Pattern: arn:[^:]+:iam::\d{12}:role/.*
Tags (p. 417)

A set of tags to associate with the S3 Batch Operations job. This is an optional parameter.

Type: Array of S3Tag (p. 726) data types

Required: No

Response Syntax

```
HTTP/1.1 200
<?xml version="1.0" encoding="UTF-8"?>
<CreateJobResult>
  <JobId>string</JobId>
</CreateJobResult>
```

Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in XML format by the service.

CreateJobResult (p. 421)

Root level tag for the CreateJobResult parameters.

Required: Yes

JobId (p. 421)

The ID for this job. Amazon S3 generates this ID automatically and returns it after a successful Create Job request.

Type: String


Pattern: [a-zA-Z0-9\-_]+

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3
DeleteAccessPoint
Service: AWS S3 Control

Deletes the specified access point.

All Amazon S3 on Outposts REST API requests for this action require an additional parameter of x-amz-outpost-id to be passed with the request and an S3 on Outposts endpoint hostname prefix instead of s3-control. For an example of the request syntax for Amazon S3 on Outposts that uses the S3 on Outposts endpoint hostname prefix and the x-amz-outpost-id derived using the access point ARN, see the Examples section.

The following actions are related to DeleteAccessPoint:
- CreateAccessPoint
- GetAccessPoint
- ListAccessPoints

Request Syntax

```
DELETE /v20180820/accesspoint/name HTTP/1.1
x-amz-account-id: AccountId
```

URI Request Parameters

The request uses the following URI parameters.

**name (p. 422)**

The name of the access point you want to delete.

For using this parameter with Amazon S3 on Outposts with the REST API, you must specify the name and the x-amz-outpost-id as well.

For using this parameter with S3 on Outposts with the AWS SDK and CLI, you must specify the ARN of the access point accessed in the format arn:aws:s3-outposts:<Region>:<account-id>:outpost/<outpost-id>/accesspoint/<my-accesspoint-name>. For example, to access the access point reports-ap through outpost my-outpost owned by account 123456789012 in Region us-west-2, use the URL encoding of arn:aws:s3-outposts:us-west-2:123456789012:outpost/my-outpost/accesspoint/reports-ap. The value must be URL encoded.


Required: Yes

**x-amz-account-id (p. 422)**

The account ID for the account that owns the specified access point.

Length Constraints: Maximum length of 64.

Pattern: ^\d{12}$

Required: Yes

Request Body

The request does not have a request body.
Response Syntax

HTTP/1.1 200

Response Elements

If the action is successful, the service sends back an HTTP 200 response with an empty HTTP body.

Examples

DeleteAccessPoint syntax for Amazon S3 on Outposts

The following request deletes the access point of the specified Outpost.

DELETE /v20180820/accesspoint/example-access-point HTTP/1.1
Host: s3-outposts.<Region>.amazonaws.com
Date: Wed, 28 Oct 2020 22:32:00 GMT
x-amz-account-id: example-account-id
x-amz-outpost-id: op-01ac5d28a6a232904
Authorization: authorization string

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3
DeleteAccessPointPolicy

Service: AWS S3 Control

Deletes the access point policy for the specified access point.

All Amazon S3 on Outposts REST API requests for this action require an additional parameter of `x-amz-outpost-id` to be passed with the request and an S3 on Outposts endpoint hostname prefix instead of `s3-control`. For an example of the request syntax for Amazon S3 on Outposts that uses the S3 on Outposts endpoint hostname prefix and the `x-amz-outpost-id` derived using the access point ARN, see the Examples section.

The following actions are related to DeleteAccessPointPolicy:
- PutAccessPointPolicy
- GetAccessPointPolicy

Request Syntax

```
DELETE /v20180820/accesspoint/name/policy HTTP/1.1
x-amz-account-id: AccountId
```

URI Request Parameters

The request uses the following URI parameters.

**name (p. 424)**

The name of the access point whose policy you want to delete.

For using this parameter with Amazon S3 on Outposts with the REST API, you must specify the name and the `x-amz-outpost-id` as well.

For using this parameter with S3 on Outposts with the AWS SDK and CLI, you must specify the ARN of the access point accessed in the format `arn:aws:s3-outposts:<Region>:<account-id>:outpost/<outpost-id>/accesspoint/<my-accesspoint-name>`. For example, to access the access point `reports-ap` through outpost `my-outpost` owned by account `123456789012` in Region `us-west-2`, use the URL encoding of `arn:aws:s3-outposts:us-west-2:123456789012:outpost/my-outpost/accesspoint/reports-ap`. The value must be URL encoded.


Required: Yes

**x-amz-account-id (p. 424)**

The account ID for the account that owns the specified access point.

Length Constraints: Maximum length of 64.

Pattern: `^\d{12}$`

Required: Yes

Request Body

The request does not have a request body.
Response Syntax

HTTP/1.1 200

Response Elements

If the action is successful, the service sends back an HTTP 200 response with an empty HTTP body.

Examples

Sample request syntax for using the DeleteAccessPointPolicy action with Amazon S3 on Outposts access point

This example illustrates one usage of DeleteAccessPointPolicy.

DELETE /v20180820/accesspoint/example-access-point/policy HTTP/1.1
Host: s3-outposts.<Region>.amazonaws.com
Date: Wed, 28 Oct 2020 22:32:00 GMT
Authorization: authorization string
x-amz-account-id: example-account-id
x-amz-outpost-id: op-01ac5d28a6a232904

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3
DeleteBucket

Service: AWS S3 Control

**Note**
This API operation deletes an Amazon S3 on Outposts bucket. To delete an S3 bucket, see DeleteBucket in the Amazon Simple Storage Service API.

Deletes the Amazon S3 on Outposts bucket. All objects (including all object versions and delete markers) in the bucket must be deleted before the bucket itself can be deleted. For more information, see Using Amazon S3 on Outposts in Amazon Simple Storage Service Developer Guide.

All Amazon S3 on Outposts REST API requests for this action require an additional parameter of x-amz-outpost-id to be passed with the request and an S3 on Outposts endpoint hostname prefix instead of s3-control. For an example of the request syntax for Amazon S3 on Outposts that uses the S3 on Outposts endpoint hostname prefix and the x-amz-outpost-id derived using the access point ARN, see the Examples section.

Related Resources
- CreateBucket
- GetBucket
- DeleteObject

Request Syntax

```
DELETE /v20180820/bucket/name HTTP/1.1
Host: Bucket.s3-control.amazonaws.com
x-amz-account-id: AccountId
```

URI Request Parameters

The request uses the following URI parameters.

**name (p. 426)**

Specifies the bucket being deleted.

For using this parameter with Amazon S3 on Outposts with the REST API, you must specify the name and the x-amz-outpost-id as well.

For using this parameter with S3 on Outposts with the AWS SDK and CLI, you must specify the ARN of the bucket accessed in the format arn:aws:s3-outposts:<Region>:<account-id>:outpost/<outpost-id>/bucket/<my-bucket-name>. For example, to access the bucket reports through outpost my-outpost owned by account 123456789012 in Region us-west-2, use the URL encoding of arn:aws:s3-outposts:us-west-2:123456789012:outpost/my-outpost/bucket/reports. The value must be URL encoded.


Required: Yes

**x-amz-account-id (p. 426)**

The account ID that owns the Outposts bucket.

Length Constraints: Maximum length of 64.

Pattern: ^\d{12}$
Required: Yes

Request Body

The request does not have a request body.

Response Syntax

HTTP/1.1 200

Response Elements

If the action is successful, the service sends back an HTTP 200 response with an empty HTTP body.

Examples

Sample request to delete an Amazon S3 on Outposts bucket

This request deletes the Outposts bucket named example-outpost-bucket.

```
DELETE /v20180820/bucket/example-outpost-bucket/ HTTP/1.1
Host: s3-outposts.<Region>.amazonaws.com
x-amz-outpost-id: op-01ac5d28a6a232904
x-amz-account-id:example-account-id
Date: Wed, 01 Mar 2006 12:00:00 GMT
Authorization: authorization string
```

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3
DeleteBucketLifecycleConfiguration
Service: AWS S3 Control

Note
This API action deletes an Amazon S3 on Outposts bucket's lifecycle configuration. To delete
an S3 bucket's lifecycle configuration, see DeleteBucketLifecycle in the Amazon Simple Storage
Service API.

Deletes the lifecycle configuration from the specified Outposts bucket. Amazon S3 on Outposts removes
all the lifecycle configuration rules in the lifecycle subresource associated with the bucket. Your objects
never expire, and Amazon S3 on Outposts no longer automatically deletes any objects on the basis of
rules contained in the deleted lifecycle configuration. For more information, see Using Amazon S3 on
Outposts in Amazon Simple Storage Service Developer Guide.

To use this operation, you must have permission to perform the s3-outposts:DeleteLifecycleConfiguration action. By default, the bucket owner has this
permission and the Outposts bucket owner can grant this permission to others.

All Amazon S3 on Outposts REST API requests for this action require an additional parameter of x-amz-outpost-id to be passed with the request and an S3 on Outposts endpoint hostname prefix instead of s3-control. For an example of the request syntax for Amazon S3 on Outposts that uses the S3 on Outposts endpoint hostname prefix and the x-amz-outpost-id derived using the access point ARN, see the Examples section.

For more information about object expiration, see Elements to Describe Lifecycle Actions.

Related actions include:
• PutBucketLifecycleConfiguration
• GetBucketLifecycleConfiguration

Request Syntax

DELETE /v20180820/bucket/name/lifecycleconfiguration HTTP/1.1
Host: Bucket.s3-control.amazonaws.com
x-amz-account-id: AccountId

URI Request Parameters

The request uses the following URI parameters.

name (p. 428)

Specifies the bucket.

For using this parameter with Amazon S3 on Outposts with the REST API, you must specify the name
and the x-amz-outpost-id as well.

For using this parameter with S3 on Outposts with the AWS SDK and CLI, you must specify the
ARN of the bucket accessed in the format arn:aws:s3-outposts:<Region>:<account-
id>:outpost/<outpost-id>/bucket/<my–bucket-name>. For example, to access the bucket reports through outpost my–outpost owned by account 123456789012 in Region us-west-2, use the URL encoding of arn:aws:s3-outposts:us-west-2:123456789012:outpost/my–outpost/bucket/reports. The value must be URL encoded.


Required: Yes
x-amz-account-id (p. 428)

The account ID of the lifecycle configuration to delete.

Length Constraints: Maximum length of 64.

Pattern: ^\d{12}$

Required: Yes

Request Body

The request does not have a request body.

Response Syntax

HTTP/1.1 200

Response Elements

If the action is successful, the service sends back an HTTP 200 response with an empty HTTP body.

Examples

Sample request to delete the lifecycle configuration of an Amazon S3 on Outposts bucket

This example illustrates one usage of DeleteBucketLifecycleConfiguration.

DELETE /v20180820/bucket/example-outpost-bucket/lifecycleconfiguration

Host: s3-outposts.<Region>.amazonaws.com
x-amz-outpost-id: op-01ac5d28a6a232904
x-amz-account-id:example-account-id

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3
DeleteBucketPolicy
Service: AWS S3 Control

Note
This API operation deletes an Amazon S3 on Outposts bucket policy. To delete an S3 bucket policy, see DeleteBucketPolicy in the Amazon Simple Storage Service API.

This implementation of the DELETE operation uses the policy subresource to delete the policy of a specified Amazon S3 on Outposts bucket. If you are using an identity other than the root user of the AWS account that owns the bucket, the calling identity must have the s3-outposts:DeleteBucketPolicy permissions on the specified Outposts bucket and belong to the bucket owner’s account to use this operation. For more information, see Using Amazon S3 on Outposts in Amazon Simple Storage Service Developer Guide.

If you don't have DeleteBucketPolicy permissions, Amazon S3 returns a 403 Access Denied error. If you have the correct permissions, but you're not using an identity that belongs to the bucket owner's account, Amazon S3 returns a 405 Method Not Allowed error.

Important
As a security precaution, the root user of the AWS account that owns a bucket can always use this operation, even if the policy explicitly denies the root user the ability to perform this action.

For more information about bucket policies, see Using Bucket Policies and User Policies.

All Amazon S3 on Outposts REST API requests for this action require an additional parameter of x-amz-outpost-id to be passed with the request and an S3 on Outposts endpoint hostname prefix instead of s3-control. For an example of the request syntax for Amazon S3 on Outposts that uses the S3 on Outposts endpoint hostname prefix and the x-amz-outpost-id derived using the access point ARN, see the Examples section.

The following actions are related to DeleteBucketPolicy:

- GetBucketPolicy
- PutBucketPolicy

Request Syntax

```
DELETE /v20180820/bucket/name/policy HTTP/1.1
Host: Bucket.s3-control.amazonaws.com
x-amz-account-id: AccountId
```

URI Request Parameters

The request uses the following URI parameters.

name (p. 430)

Specifies the bucket.

For using this parameter with Amazon S3 on Outposts with the REST API, you must specify the name and the x-amz-outpost-id as well.

For using this parameter with S3 on Outposts with the AWS SDK and CLI, you must specify the ARN of the bucket accessed in the format arn:aws:s3-outposts:<Region>:<account-id>:outpost/<outpost-id>/bucket/<my-bucket-name>. For example, to access the bucket reports through outpost my-outpost owned by account 123456789012 in Region us-west-2, use the URL encoding of arn:aws:s3-outposts:us-west-2:123456789012:outpost/my-outpost/bucket/reports. The value must be URL encoded.

Required: Yes

x-amz-account-id (p. 430)

The account ID of the Outposts bucket.

Length Constraints: Maximum length of 64.

Pattern: ^\d{12}$

Required: Yes

**Request Body**

The request does not have a request body.

**Response Syntax**

```
HTTP/1.1 200
```

**Response Elements**

If the action is successful, the service sends back an HTTP 200 response with an empty HTTP body.

**Examples**

Sample request for deleting a bucket policy for an Amazon S3 on Outposts bucket

This example illustrates one usage of DeleteBucketPolicy.

```
DELETE v20180820/bucket/example-outpost-bucket/policy  HTTP/1.1
Host: s3-outposts.<Region>.amazonaws.com
x-amz-account-id: example-account-id
x-amz-outpost-id: op-01ac5d28a6a232904
```

**See Also**

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3
DeleteBucketTagging
Service: AWS S3 Control

Note
This API operation deletes an Amazon S3 on Outposts bucket's tags. To delete an S3 bucket tags, see DeleteBucketTagging in the Amazon Simple Storage Service API.

Deletes the tags from the Outposts bucket. For more information, see Using Amazon S3 on Outposts in Amazon Simple Storage Service Developer Guide.

To use this operation, you must have permission to perform the PutBucketTagging action. By default, the bucket owner has this permission and can grant this permission to others.

All Amazon S3 on Outposts REST API requests for this action require an additional parameter of x-amz-outpost-id to be passed with the request and an S3 on Outposts endpoint hostname prefix instead of s3-control. For an example of the request syntax for Amazon S3 on Outposts that uses the S3 on Outposts endpoint hostname prefix and the x-amz-outpost-id derived using the access point ARN, see the Examples section.

The following actions are related to DeleteBucketTagging:
- GetBucketTagging
- PutBucketTagging

Request Syntax

DELETE /v20180820/bucket/name/tagging HTTP/1.1
Host: Bucket.s3-control.amazonaws.com
x-amz-account-id: AccountId

URI Request Parameters

The request uses the following URI parameters.

name (p. 432)
The bucket ARN that has the tag set to be removed.

For using this parameter with Amazon S3 on Outposts with the REST API, you must specify the name and the x-amz-outpost-id as well.

For using this parameter with S3 on Outposts with the AWS SDK and CLI, you must specify the ARN of the bucket accessed in the format arn:aws:s3-outposts:<Region>;<account-id>:outpost/<outpost-id>/bucket/<my-bucket-name>. For example, to access the bucket reports through outpost my-outpost owned by account 123456789012 in Region us-west-2, use the URL encoding of arn:aws:s3-outposts:us-west-2:123456789012:outpost/my-outpost/bucket/reports. The value must be URL encoded.


Required: Yes

x-amz-account-id (p. 432)
The AWS account ID of the Outposts bucket tag set to be removed.

Length Constraints: Maximum length of 64.

Pattern: ^\d{12}$
Required: Yes

Request Body

The request does not have a request body.

Response Syntax

```
HTTP/1.1 204
```

Response Elements

If the action is successful, the service sends back an HTTP 204 response with an empty HTTP body.

Examples

Sample request to delete tags for Amazon S3 on Outposts bucket

The following DELETE request deletes the tag set from the Outposts bucket `example-outpost-bucket`.

```
DELETE v20180820/bucket/example-outpost-bucket/tagging HTTP/1.1
Host: s3-outposts.<Region>.amazonaws.com
x-amz-account-id: example-account-id
x-amz-outpost-id: op-01ac5d28a6a232904
Date: Wed, 14 Dec 2020 05:37:16 GMT
Authorization: signatureValue
```

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS Command Line Interface](#)
- [AWS SDK for .NET](#)
- [AWS SDK for C++](#)
- [AWS SDK for Go](#)
- [AWS SDK for Java](#)
- [AWS SDK for JavaScript](#)
- [AWS SDK for PHP V3](#)
- [AWS SDK for Python](#)
- [AWS SDK for Ruby V3](#)
**DeleteJobTagging**

Service: AWS S3 Control

Removes the entire tag set from the specified S3 Batch Operations job. To use this operation, you must have permission to perform the s3:DeleteJobTagging action. For more information, see Controlling access and labeling jobs using tags in the Amazon Simple Storage Service Developer Guide.

Related actions include:

- CreateJob
- GetJobTagging
- PutJobTagging

**Request Syntax**

```
DELETE /v20180820/jobs/id/tagging HTTP/1.1
x-amz-account-id: AccountId
```

**URI Request Parameters**

The request uses the following URI parameters.

- **id** *(p. 434)*
  
The ID for the S3 Batch Operations job whose tags you want to delete.
  
  
  Pattern: `[a-zA-Z0-9\-\_]+`
  
  Required: Yes

- **x-amz-account-id** *(p. 434)*
  
The AWS account ID associated with the S3 Batch Operations job.
  
  Length Constraints: Maximum length of 64.
  
  Pattern: `^[\d]{12}$`
  
  Required: Yes

**Request Body**

The request does not have a request body.

**Response Syntax**

```
HTTP/1.1 200
```

**Response Elements**

If the action is successful, the service sends back an HTTP 200 response with an empty HTTP body.

**See Also**

For more information about using this API in one of the language-specific AWS SDKs, see the following:
• AWS Command Line Interface
• AWS SDK for .NET
• AWS SDK for C++
• AWS SDK for Go
• AWS SDK for Java
• AWS SDK for JavaScript
• AWS SDK for PHP V3
• AWS SDK for Python
• AWS SDK for Ruby V3
DeletePublicAccessBlock
Service: AWS S3 Control

Removes the PublicAccessBlock configuration for an AWS account. For more information, see Using Amazon S3 block public access.

Related actions include:
• GetPublicAccessBlock
• PutPublicAccessBlock

Request Syntax

```
DELETE /v20180820/configuration/publicAccessBlock HTTP/1.1
x-amz-account-id: AccountId
```

URI Request Parameters

The request uses the following URI parameters.

`x-amz-account-id (p. 436)`

The account ID for the AWS account whose PublicAccessBlock configuration you want to remove.

Length Constraints: Maximum length of 64.

Pattern: `^\d{12}$`

Required: Yes

Request Body

The request does not have a request body.

Response Syntax

```
HTTP/1.1 200
```

Response Elements

If the action is successful, the service sends back an HTTP 200 response with an empty HTTP body.

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

• AWS Command Line Interface
• AWS SDK for .NET
• AWS SDK for C++
• AWS SDK for Go
• AWS SDK for Java
• AWS SDK for JavaScript

API Version 2006-03-01
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3
DescribeJob

Service: AWS S3 Control

Retrieves the configuration parameters and status for a Batch Operations job. For more information, see S3 Batch Operations in the Amazon Simple Storage Service Developer Guide.

Related actions include:
- CreateJob
- ListJobs
- UpdateJobPriority
- UpdateJobStatus

Request Syntax

GET /v20180820/jobs/id HTTP/1.1
x-amz-account-id: AccountId

URI Request Parameters

The request uses the following URI parameters.

id (p. 438)

The ID for the job whose information you want to retrieve.


Pattern: [a-zA-Z0-9\-\_\.]*

Required: Yes

x-amz-account-id (p. 438)

Length Constraints: Maximum length of 64.

Pattern: ^\d{12}$

Required: Yes

Request Body

The request does not have a request body.

Response Syntax

HTTP/1.1 200
<?xml version="1.0" encoding="UTF-8"?>
<DescribeJobResult>
  <Job>
    <ConfirmationRequired>boolean</ConfirmationRequired>
    <CreationTime>timestamp</CreationTime>
    <Description>string</Description>
    <FailureReasons>
      <JobFailure>
        <FailureCode>string</FailureCode>
        <FailureReason>string</FailureReason>
      </JobFailure>
    </FailureReasons>
  </Job>
</DescribeJobResult>
<JobFailure>
</FailureReasons>
<JobArn>string</JobArn>
<JobId>string</JobId>
<Manifest>
<Location>
<ETag>string</ETag>
<ObjectArn>string</ObjectArn>
<ObjectVersionId>string</ObjectVersionId>
</Location>
</Spec>
<Fields>
<INVALID-TYPE-NAME>string</INVALID-TYPE-NAME>
</Fields>
</Format>string</Format>
</Spec>
</Manifest>
<Operation>
<LambdaInvoke>
<FunctionArn>string</FunctionArn>
</LambdaInvoke>
<S3InitiateRestoreObject>
<ExpirationInDays>integer</ExpirationInDays>
<GlacierJobTier>string</GlacierJobTier>
</S3InitiateRestoreObject>
<S3PutObjectAcl>
<AccessControlPolicy>
<AccessControlList>
<Grants>
<S3Grant>
<Grantee>
<DisplayName>string</DisplayName>
<Identifier>string</Identifier>
<TypeIdentifier>string</TypeIdentifier>
</Grantee>
<Permission>string</Permission>
</S3Grant>
</Grants>
<Owner>
<DisplayName>string</DisplayName>
<ID>string</ID>
</Owner>
</AccessControlList>
</AccessControlPolicy>
</S3PutObjectAcl>
<S3PutObjectCopy>
<AccessControlGrants>
<S3Grant>
<Grantee>
<DisplayName>string</DisplayName>
<Identifier>string</Identifier>
<TypeIdentifier>string</TypeIdentifier>
</Grantee>
<Permission>string</Permission>
</S3Grant>
</AccessControlGrants>
</S3PutObjectCopy>
</LambdaInvoke>
</Operation>
</LambdaInvoke>
</S3InitiateRestoreObject>
</S3PutObjectAcl>
</S3PutObjectCopy>
</LambdaInvoke>
</Operation>
</API Version 2006-03-01
439
<ContentMD5>string</ContentMD5>
<ContentType>string</ContentType>
<HttpExpiresDate>timestamp</HttpExpiresDate>
<RequesterCharged>boolean</RequesterCharged>
<SSEAlgorithm>string</SSEAlgorithm>
<UserMetadata>
  <entry>
    <key>string</key>
    <value>string</value>
  </entry>
</UserMetadata>
</NewObjectMetadata>

<NewObjectTagging>
  <Tag>
    <Key>string</Key>
    <Value>string</Value>
  </Tag>
</NewObjectTagging>

<ObjectLockLegalHoldStatus>string</ObjectLockLegalHoldStatus>
<ObjectLockMode>string</ObjectLockMode>
<ObjectLockRetainUntilDate>timestamp</ObjectLockRetainUntilDate>
<RedirectLocation>string</RedirectLocation>
<RequesterPays>boolean</RequesterPays>
<SSEAwsKmsKeyId>string</SSEAwsKmsKeyId>
<StorageClass>string</StorageClass>
<TargetKeyPrefix>string</TargetKeyPrefix>
<TargetResource>string</TargetResource>
<UnModifiedSinceConstraint>timestamp</UnModifiedSinceConstraint>
</S3PutObjectCopy>

<S3PutObjectLegalHold>
  <LegalHold>
    <Status>string</Status>
  </LegalHold>
</S3PutObjectLegalHold>

<BypassGovernanceRetention>boolean</BypassGovernanceRetention>
<Retention>
  <Mode>string</Mode>
  <RetainUntilDate>timestamp</RetainUntilDate>
</Retention>

<S3PutObjectRetention>
  <TagSet>
    <Tag>
      <Key>string</Key>
      <Value>string</Value>
    </Tag>
  </TagSet>
</S3PutObjectRetention>

<S3PutObjectTagging>
  <TagSet>
    <Tag>
      <Key>string</Key>
      <Value>string</Value>
    </Tag>
  </TagSet>
</S3PutObjectTagging>

<Operation>
  <Priority>integer</Priority>
</Operation>

<ProgressSummary>
  <NumberOfTasksFailed>long</NumberOfTasksFailed>
  <NumberOfTasksSucceeded>long</NumberOfTasksSucceeded>
  <TotalNumberOfTasks>long</TotalNumberOfTasks>
</ProgressSummary>

<Report>
  <Bucket>string</Bucket>
  <Enabled>boolean</Enabled>
  <Format>string</Format>
  <Prefix>string</Prefix>
  <ReportScope>string</ReportScope>
</Report>

<RoleArn>string</RoleArn>
>Status>string</Status>
</StatusUpdateReason>
Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in XML format by the service.

**DescribeJobResult (p. 438)**

Root level tag for the DescribeJobResult parameters.

Required: Yes

**Job (p. 438)**

Contains the configuration parameters and status for the job specified in the Describe Job request.

Type: JobDescriptor (p. 682) data type

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3
GetAccessPoint
Service: AWS S3 Control

Returns configuration information about the specified access point.

All Amazon S3 on Outposts REST API requests for this action require an additional parameter of x-amz-outpost-id to be passed with the request and an S3 on Outposts endpoint hostname prefix instead of s3-control. For an example of the request syntax for Amazon S3 on Outposts that uses the S3 on Outposts endpoint hostname prefix and the x-amz-outpost-id derived using the access point ARN, see the Examples section.

The following actions are related to GetAccessPoint:
- CreateAccessPoint
- DeleteAccessPoint
- ListAccessPoints

Request Syntax

```
GET /v20180820/accesspoint/name HTTP/1.1
x-amz-account-id: AccountId
```

URI Request Parameters

The request uses the following URI parameters.

name (p. 442)

The name of the access point whose configuration information you want to retrieve.

For using this parameter with Amazon S3 on Outposts with the REST API, you must specify the name and the x-amz-outpost-id as well.

For using this parameter with S3 on Outposts with the AWS SDK and CLI, you must specify the ARN of the access point accessed in the format arn:aws:s3-outposts:<Region>:<account-id>:outpost/<outpost-id>/accesspoint/<my-accesspoint-name>. For example, to access the access point reports-ap through outpost my-outpost owned by account 123456789012 in Region us-west-2, use the URL encoding of arn:aws:s3-outposts:us-west-2:123456789012:outpost/my-outpost/accesspoint/reports-ap. The value must be URL encoded.


Required: Yes

x-amz-account-id (p. 442)

The account ID for the account that owns the specified access point.

Length Constraints: Maximum length of 64.

Pattern: ^\d{12}$

Required: Yes

Request Body

The request does not have a request body.
Response Syntax

HTTP/1.1 200
<?xml version="1.0" encoding="UTF-8"?>
<GetAccessPointResult>
  <Name>string</Name>
  <Bucket>string</Bucket>
  <NetworkOrigin>string</NetworkOrigin>
  <VpcConfiguration>
    <VpcId>string</VpcId>
  </VpcConfiguration>
  <PublicAccessBlockConfiguration>
    <BlockPublicAcls>boolean</BlockPublicAcls>
    <BlockPublicPolicy>boolean</BlockPublicPolicy>
    <IgnorePublicAcls>boolean</IgnorePublicAcls>
    <RestrictPublicBuckets>boolean</RestrictPublicBuckets>
  </PublicAccessBlockConfiguration>
  <CreationDate>timestamp</CreationDate>
</GetAccessPointResult>

Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in XML format by the service.

**GetAccessPointResult (p. 443)**

Root level tag for the GetAccessPointResult parameters.

Required: Yes

**Bucket (p. 443)**

The name of the bucket associated with the specified access point.

Type: String


**CreationDate (p. 443)**

The date and time when the specified access point was created.

Type: Timestamp

**Name (p. 443)**

The name of the specified access point.

Type: String


**NetworkOrigin (p. 443)**

Indicates whether this access point allows access from the public internet. If VpcConfiguration is specified for this access point, then NetworkOrigin is VPC, and the access point doesn't allow access from the public internet. Otherwise, NetworkOrigin is Internet, and the access point allows access from the public internet, subject to the access point and bucket access policies.

This will always be true for an Amazon S3 on Outposts access point.

Type: String
Valid Values: Internet | VPC

PublicAccessBlockConfiguration (p. 443)

The PublicAccessBlock configuration that you want to apply to this Amazon S3 account. You can enable the configuration options in any combination. For more information about when Amazon S3 considers a bucket or object public, see The Meaning of "Public" in the Amazon Simple Storage Service Developer Guide.

This is not supported for Amazon S3 on Outposts.

Type: PublicAccessBlockConfiguration (p. 706) data type

VpcConfiguration (p. 443)

Contains the virtual private cloud (VPC) configuration for the specified access point.

Type: VpcConfiguration (p. 729) data type

Examples

Sample request syntax for getting an Amazon S3 on Outposts access point

The following request returns the access point of the specified S3 on Outposts access point.

```plaintext
GET /v20180820/accesspoint/example-access-point HTTP/1.1
Host: s3-outposts.<Region>.amazonaws.com
Date: Wed, 28 Oct 2020 22:32:00 GMT
Authorization: authorization string
x-amz-account-id: example-account-id
x-amz-outpost-id: op-01ac5d28a6a232904
```

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3
GetAccessPointPolicy
Service: AWS S3 Control

Returns the access point policy associated with the specified access point.

The following actions are related to GetAccessPointPolicy:

- PutAccessPointPolicy
- DeleteAccessPointPolicy

Request Syntax

GET /v20180820/accesspoint/name/policy HTTP/1.1
x-amz-account-id: AccountId

URI Request Parameters

The request uses the following URI parameters.

name (p. 445)

The name of the access point whose policy you want to retrieve.

For using this parameter with Amazon S3 on Outposts with the REST API, you must specify the name and the x-amz-outpost-id as well.

For using this parameter with S3 on Outposts with the AWS SDK and CLI, you must specify the ARN of the access point accessed in the format arn:aws:s3-outposts:<Region>:<account-id>:outpost/<outpost-id>/accesspoint/<my-accesspoint-name>. For example, to access the access point reports-ap through outpost my-outpost owned by account 123456789012 in Region us-west-2, use the URL encoding of arn:aws:s3-outposts:us-west-2:123456789012:outpost/my-outpost/accesspoint/reports-ap. The value must be URL encoded.


Required: Yes

x-amz-account-id (p. 445)

The account ID for the account that owns the specified access point.

Length Constraints: Maximum length of 64.

Pattern: ^\d{12}$

Required: Yes

Request Body

The request does not have a request body.

Response Syntax

HTTP/1.1 200
<?xml version="1.0" encoding="UTF-8"?>
<GetAccessPointPolicyResult>
Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in XML format by the service.

GetAccessPointPolicyResult (p. 445)

Root level tag for the GetAccessPointPolicyResult parameters.

Required: Yes

Policy (p. 445)

The access point policy associated with the specified access point.

Type: String

Examples

Sample request

The following request returns the access point of the specified Amazon S3 on Outposts.

```
GET /v20180820/accesspoint/example-access-point/policy HTTP/1.1
Host: s3-outposts.<Region>.amazonaws.com
Date: Wed, 28 Oct 2020 22:32:00 GMT
Authorization: authorization string
x-amz-account-id: 123456789012
x-amz-outpost-id: op-123456
```

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3
GetAccessPointPolicyStatus
Service: AWS S3 Control

Indicates whether the specified access point currently has a policy that allows public access. For more information about public access through access points, see Managing Data Access with Amazon S3 Access Points in the Amazon Simple Storage Service Developer Guide.

Request Syntax

GET /v20180820/accesspoint/name/policyStatus HTTP/1.1
x-amz-account-id: AccountId

URI Request Parameters

The request uses the following URI parameters.

name (p. 447)
The name of the access point whose policy status you want to retrieve.
Required: Yes

x-amz-account-id (p. 447)
The account ID for the account that owns the specified access point.
Length Constraints: Maximum length of 64.
Pattern: ^\d{12}$
Required: Yes

Request Body

The request does not have a request body.

Response Syntax

HTTP/1.1 200
<?xml version="1.0" encoding="UTF-8"?>
<GetAccessPointPolicyStatusResult>
  <PolicyStatus>
    <IsPublic>boolean</IsPublic>
  </PolicyStatus>
</GetAccessPointPolicyStatusResult>

Response Elements

If the action is successful, the service sends back an HTTP 200 response.
The following data is returned in XML format by the service.

GetAccessPointPolicyStatusResult (p. 447)

Root level tag for the GetAccessPointPolicyStatusResult parameters.
Required: Yes

**PolicyStatus (p. 447)**

Indicates the current policy status of the specified access point.

Type: **PolicyStatus (p. 705)** data type

**See Also**

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS Command Line Interface](#)
- [AWS SDK for .NET](#)
- [AWS SDK for C++](#)
- [AWS SDK for Go](#)
- [AWS SDK for Java](#)
- [AWS SDK for JavaScript](#)
- [AWS SDK for PHP V3](#)
- [AWS SDK for Python](#)
- [AWS SDK for Ruby V3](#)
GetBucket
Service: AWS S3 Control

Gets an Amazon S3 on Outposts bucket. For more information, see Using Amazon S3 on Outposts in the Amazon Simple Storage Service Developer Guide.

If you are using an identity other than the root user of the AWS account that owns the bucket, the calling identity must have the s3-outposts:GetBucket permissions on the specified bucket and belong to the bucket owner's account in order to use this operation. Only users from Outposts bucket owner account with the right permissions can perform actions on an Outposts bucket.

If you don't have s3-outposts:GetBucket permissions or you're not using an identity that belongs to the bucket owner's account, Amazon S3 returns a 403 Access Denied error.

The following actions are related to GetBucket for Amazon S3 on Outposts:

- PutObject
- CreateBucket
- DeleteBucket

Request Syntax

```
GET /v20180820/bucket/name HTTP/1.1
Host: Bucket.s3-control.amazonaws.com
x-amz-account-id: AccountId
```

URI Request Parameters

The request uses the following URI parameters.

name (p. 449)

Specifies the bucket.

For using this parameter with Amazon S3 on Outposts with the REST API, you must specify the name and the x-amz-outpost-id as well.

For using this parameter with S3 on Outposts with the AWS SDK and CLI, you must specify the ARN of the bucket accessed in the format arn:aws:s3-outposts:<Region>:<account-id>:outpost/<outpost-id>/bucket/<my-bucket-name>. For example, to access the bucket reports through outpost my-outpost owned by account 123456789012 in Region us-west-2, use the URL encoding of arn:aws:s3-outposts:us-west-2:123456789012:outpost/my-outpost/bucket/reports. The value must be URL encoded.


Required: Yes

x-amz-account-id (p. 449)

The AWS account ID of the Outposts bucket.
Length Constraints: Maximum length of 64.

Pattern: ^\d{12}$

Required: Yes

**Request Body**

The request does not have a request body.

**Response Syntax**

```xml
HTTP/1.1 200
<?xml version="1.0" encoding="UTF-8"?>
<GetBucketResult>
    <Bucket>string</Bucket>
    <PublicAccessBlockEnabled>boolean</PublicAccessBlockEnabled>
    <CreationDate>timestamp</CreationDate>
</GetBucketResult>
```

**Response Elements**

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in XML format by the service.

**GetBucketResult** (p. 450)

Root level tag for the GetBucketResult parameters.

Required: Yes

**Bucket** (p. 450)

The Outposts bucket requested.

Type: String


**CreationDate** (p. 450)

The creation date of the Outposts bucket.

Type: Timestamp

**PublicAccessBlockEnabled** (p. 450)

Type: Boolean

**Examples**

**Sample request for getting Amazon S3 on Outposts bucket**

This example illustrates one usage of GetBucket.

```
GET /v20180820/bucket/example-outpost-bucket/ HTTP/1.1
Host: s3-outposts.<Region>.amazonaws.com
x-amz-account-id: example-account-id
x-amz-outpost-id: op-01ac5d28a6a232904
```
See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3
GetBucketLifecycleConfiguration
Service: AWS S3 Control

Note
This API operation gets an Amazon S3 on Outposts bucket's lifecycle configuration. To get an
S3 bucket's lifecycle configuration, see GetBucketLifecycleConfiguration in the Amazon Simple
Storage Service API.

Returns the lifecycle configuration information set on the Outposts bucket. For more information, see
Using Amazon S3 on Outposts and for information about lifecycle configuration, see Object Lifecycle
Management in Amazon Simple Storage Service Developer Guide.

To use this operation, you must have permission to perform the s3-outposts:GetLifecycleConfiguration
action. The Outposts bucket owner has this permission, by
default. The bucket owner can grant this permission to others. For more information about permissions,
see Permissions Related to Bucket Subresource Operations and Managing Access Permissions to Your
Amazon S3 Resources.

All Amazon S3 on Outposts REST API requests for this action require an additional parameter of
x-amz-outpost-id to be passed with the request and an S3 on Outposts endpoint hostname prefix instead
of s3-control. For an example of the request syntax for Amazon S3 on Outposts that uses the S3 on
Outposts endpoint hostname prefix and the x-amz-outpost-id derived using the access point ARN,
see the Examples section.

GetBucketLifecycleConfiguration has the following special error:

- Error code: NoSuchLifecycleConfiguration
- Description: The lifecycle configuration does not exist.
- HTTP Status Code: 404 Not Found
- SOAP Fault Code Prefix: Client

The following actions are related to GetBucketLifecycleConfiguration:

- PutBucketLifecycleConfiguration
- DeleteBucketLifecycleConfiguration

Request Syntax

```
GET /v20180820/bucket/name/lifecycleconfiguration HTTP/1.1
Host: Bucket.s3-control.amazonaws.com
x-amz-account-id: AccountId
```

URI Request Parameters

The request uses the following URI parameters.

name (p. 452)

The Amazon Resource Name (ARN) of the bucket.

For using this parameter with Amazon S3 on Outposts with the REST API, you must specify the name
and the x-amz-outpost-id as well.

For using this parameter with S3 on Outposts with the AWS SDK and CLI, you must specify the
ARN of the bucket accessed in the format arn:aws:s3-outposts:<Region>:<account-id>:outpost:<outpost-id>/bucket/<my-bucket-name>. For example, to access the bucket
reports through outpost my-outpost owned by account 123456789012 in Region us-west-2, use the URL encoding of arn:aws:s3-outposts:us-west-2:123456789012:outpost/my-outpost/bucket/reports. The value must be URL encoded.


Required: Yes

x-amz-account-id (p. 452)

The AWS account ID of the Outposts bucket.

Length Constraints: Maximum length of 64.

Pattern: ^\d{12}$

Required: Yes

Request Body

The request does not have a request body.

Response Syntax

HTTP/1.1 200
<?xml version="1.0" encoding="UTF-8"?>
<GetBucketLifecycleConfigurationResult>
  <Rules>
    <Rule>
      <AbortIncompleteMultipartUpload>
        <DaysAfterInitiation>integer</DaysAfterInitiation>
      </AbortIncompleteMultipartUpload>
      <Expiration>
        <Date>timestamp</Date>
        <Days>integer</Days>
        <ExpiredObjectDeleteMarker>boolean</ExpiredObjectDeleteMarker>
      </Expiration>
      <Filter>
        <And>
          <Prefix>string</Prefix>
          <Tags>
            <Tag>
              <Key>string</Key>
              <Value>string</Value>
            </Tag>
          </Tags>
        </And>
        <Prefix>string</Prefix>
        <Tag>
          <Key>string</Key>
          <Value>string</Value>
        </Tag>
      </Filter>
      <ID>string</ID>
      <NoncurrentVersionExpiration>
        <NoncurrentDays>integer</NoncurrentDays>
      </NoncurrentVersionExpiration>
      <NoncurrentVersionTransitions>
        <NoncurrentDays>integer</NoncurrentDays>
        <StorageClass>string</StorageClass>
      </NoncurrentVersionTransitions>
    </Rule>
  </Rules>
</GetBucketLifecycleConfigurationResult>
Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in XML format by the service.

**GetBucketLifecycleConfigurationResult (p. 453)**

Root level tag for the GetBucketLifecycleConfigurationResult parameters.

Required: Yes

**Rules (p. 453)**

Container for the lifecycle rule of the Outposts bucket.

Type: Array of LifecycleRule (p. 699) data types

Examples

Sample request to get the lifecycle configuration of the Amazon S3 on Outposts bucket

The following example shows how to get the lifecycle configuration of the Outposts bucket.

```
GET /v20180820/bucket/example-outpost-bucket/lifecycleconfiguration HTTP/1.1
Host: s3-outposts.<Region>.amazonaws.com
x-amz-account-id: example-account-id
x-amz-outpost-id: op-01ac5d28a6a232904
x-amz-date: Thu, 15 Nov 2012 00:17:21 GMT
Authorization: signatureValue
```

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3
GetBucketPolicy
Service: AWS S3 Control

Note
This API action gets a bucket policy for an Amazon S3 on Outposts bucket. To get a policy for an
S3 bucket, see GetBucketPolicy in the Amazon Simple Storage Service API.

Returns the policy of a specified Outposts bucket. For more information, see Using Amazon S3 on
Outposts in the Amazon Simple Storage Service Developer Guide.

If you are using an identity other than the root user of the AWS account that owns the bucket, the calling
identity must have the GetBucketPolicy permissions on the specified bucket and belong to the bucket
owner's account in order to use this operation.

Only users from Outposts bucket owner account with the right permissions can perform actions on an
Outposts bucket. If you don't have s3-outposts:GetBucketPolicy permissions or you're not using
an identity that belongs to the bucket owner's account, Amazon S3 returns a 403 Access Denied error.

Important
As a security precaution, the root user of the AWS account that owns a bucket can always use
this operation, even if the policy explicitly denies the root user the ability to perform this action.

For more information about bucket policies, see Using Bucket Policies and User Policies.

All Amazon S3 on Outposts REST API requests for this action require an additional parameter of x-amz-
outpost-id to be passed with the request and an S3 on Outposts endpoint hostname prefix instead
of s3-control. For an example of the request syntax for Amazon S3 on Outposts that uses the S3 on
Outposts endpoint hostname prefix and the x-amz-outpost-id derived using the access point ARN,
see the Examples section.

The following actions are related to GetBucketPolicy:

- GetObject
- PutBucketPolicy
- DeleteBucketPolicy

Request Syntax

GET /v20180820/bucket/name/policy HTTP/1.1
Host: Bucket.s3-control.amazonaws.com
x-amz-account-id: AccountId

URI Request Parameters

The request uses the following URI parameters.

name (p. 456)

Specifies the bucket.

For using this parameter with Amazon S3 on Outposts with the REST API, you must specify the name
and the x-amz-outpost-id as well.

For using this parameter with S3 on Outposts with the AWS SDK and CLI, you must specify the
ARN of the bucket accessed in the format arn:aws:s3-outposts:<Region>:<account-
id>:outpost:<outpost-id>/bucket/<my-bucket-name>. For example, to access the bucket
reports through outpost my-outpost owned by account 123456789012 in Region us-west-2,
use the URL encoding of `arn:aws:s3-outposts:us-west-2:123456789012:outpost/my-outpost/bucket/reports`. The value must be URL encoded.


Required: Yes

**x-amz-account-id (p. 456)**

The AWS account ID of the Outposts bucket.

Length Constraints: Maximum length of 64.

Pattern: `^\d{12}$`

Required: Yes

**Request Body**

The request does not have a request body.

**Response Syntax**

<table>
<thead>
<tr>
<th>HTTP/1.1 200</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>&lt;?xml version=&quot;1.0&quot; encoding=&quot;UTF-8&quot;?&gt;</code></td>
</tr>
<tr>
<td><code>&lt;GetBucketPolicyResult&gt;</code></td>
</tr>
<tr>
<td><code>  &lt;Policy&gt;</code></td>
</tr>
<tr>
<td><code>    &lt;string&gt;</code></td>
</tr>
<tr>
<td><code>&lt;/Policy&gt;</code></td>
</tr>
<tr>
<td><code>&lt;/GetBucketPolicyResult&gt;</code></td>
</tr>
</tbody>
</table>

**Response Elements**

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in XML format by the service.

**GetBucketPolicyResult (p. 457)**

Root level tag for the GetBucketPolicyResult parameters.

Required: Yes

**Policy (p. 457)**

The policy of the Outposts bucket.

Type: String

**Examples**

**Sample GetBucketPolicy request for an Amazon S3 on Outposts bucket**

The following request gets the policy of the specified Outposts bucket example-outpost-bucket.

```
GET /v20180820/bucket/example-outpost-bucket/policy HTTP/1.1
Host: s3-outposts.<Region>.amazonaws.com
Date: Wed, 28 Oct 2009 22:32:00 GMT
Authorization: authorization string
x-amz-account-id: example-account-id
x-amz-outpost-id: op-01ac5d28a6a232904
```
See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3
GetBucketTagging
Service: AWS S3 Control

Note
This API operation gets an Amazon S3 on Outposts bucket's tags. To get an S3 bucket tags, see GetBucketTagging in the Amazon Simple Storage Service API.

Returns the tag set associated with the Outposts bucket. For more information, see Using Amazon S3 on Outposts in the Amazon Simple Storage Service Developer Guide.

To use this operation, you must have permission to perform the GetBucketTagging action. By default, the bucket owner has this permission and can grant this permission to others.

GetBucketTagging has the following special error:

- Error code: NoSuchTagSetError
- Description: There is no tag set associated with the bucket.

All Amazon S3 on Outposts REST API requests for this action require an additional parameter of x-amz-outpost-id to be passed with the request and an S3 on Outposts endpoint hostname prefix instead of s3-control. For an example of the request syntax for Amazon S3 on Outposts that uses the S3 on Outposts endpoint hostname prefix and the x-amz-outpost-id derived using the access point ARN, see the Examples section.

The following actions are related to GetBucketTagging:
- PutBucketTagging
- DeleteBucketTagging

Request Syntax

GET /v20180820/bucket/name/tagging HTTP/1.1
Host: Bucket.s3-control.amazonaws.com
x-amz-account-id: AccountId

URI Request Parameters

The request uses the following URI parameters.

name (p. 459)

Specifies the bucket.

For using this parameter with Amazon S3 on Outposts with the REST API, you must specify the name and the x-amz-outpost-id as well.

For using this parameter with S3 on Outposts with the AWS SDK and CLI, you must specify the ARN of the bucket accessed in the format arn:aws:s3-outposts:<Region>:{account-id}:outpost/<outpost-id>/bucket/<my-bucket-name>. For example, to access the bucket reports through outpost my-outpost owned by account 123456789012 in Region us-west-2, use the URL encoding of arn:aws:s3-outposts:us-west-2:123456789012:outpost/my-outpost/bucket/reports. The value must be URL encoded.


Required: Yes
x-amz-account-id (p. 459)

The AWS account ID of the Outposts bucket.

Length Constraints: Maximum length of 64.

Pattern: ^\d{12}$

Required: Yes

Request Body

The request does not have a request body.

Response Syntax

HTTP/1.1 200
<?xml version="1.0" encoding="UTF-8"?>
<GetBucketTaggingResult>
  <TagSet>
    <Tag>
      <Key>string</Key>
      <Value>string</Value>
    </Tag>
  </TagSet>
</GetBucketTaggingResult>

Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in XML format by the service.

GetBucketTaggingResult (p. 460)

Root level tag for the GetBucketTaggingResult parameters.

Required: Yes

TagSet (p. 460)

The tags set of the Outposts bucket.

Type: Array of S3Tag (p. 726) data types

Examples

Amazon S3 on Outposts request example for getting a tag set for an Outposts bucket

The following request gets the tag set of the specified Outposts bucket example-outpost-bucket.

```
GET /v20180820/bucket/example-outpost-bucket/tagging HTTP/1.1
Host: s3-outposts.<Region>.amazonaws.com
x-amz-date: Wed, 28 Oct 2020 22:32:00 GMT
x-amz-account-id: example-account-id
x-amz-outpost-id: op-01ac5d28a6a232904
Authorization: authorization string
```
See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3
GetJobTagging
Service: AWS S3 Control

Returns the tags on an S3 Batch Operations job. To use this operation, you must have permission to perform the s3:GetJobTagging action. For more information, see Controlling access and labeling jobs using tags in the Amazon Simple Storage Service Developer Guide.

Related actions include:
• CreateJob
• PutJobTagging
• DeleteJobTagging

Request Syntax

GET /v20180820/jobs/id/tagging HTTP/1.1
x-amz-account-id: AccountId

URI Request Parameters

The request uses the following URI parameters.

id (p. 462)

The ID for the S3 Batch Operations job whose tags you want to retrieve.


Pattern: [a-zA-Z0-9\-\_]+

Required: Yes

x-amz-account-id (p. 462)

The AWS account ID associated with the S3 Batch Operations job.

Length Constraints: Maximum length of 64.

Pattern: ^\d{12}$

Required: Yes

Request Body

The request does not have a request body.

Response Syntax

HTTP/1.1 200
<?xml version="1.0" encoding="UTF-8"?>
<GetJobTaggingResult>
  <Tags>
    <Tag>
      <Key>string</Key>
      <Value>string</Value>
    </Tag>
  </Tags>
</GetJobTaggingResult>
Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in XML format by the service.

**GetJobTaggingResult (p. 462)**

Root level tag for the GetJobTaggingResult parameters.

Required: Yes

**Tags (p. 462)**

The set of tags associated with the S3 Batch Operations job.

Type: Array of [S3Tag (p. 726)] data types

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3
GetPublicAccessBlock
Service: AWS S3 Control

Retrieves the PublicAccessBlock configuration for an AWS account. For more information, see Using Amazon S3 block public access.

Related actions include:
- DeletePublicAccessBlock
- PutPublicAccessBlock

Request Syntax

GET /v20180820/configuration/publicAccessBlock HTTP/1.1
x-amz-account-id: AccountId

URI Request Parameters

The request uses the following URI parameters.

x-amz-account-id (p. 464)

The account ID for the AWS account whose PublicAccessBlock configuration you want to retrieve.

Length Constraints: Maximum length of 64.
Pattern: ^\d{12}$
Required: Yes

Request Body

The request does not have a request body.

Response Syntax

HTTP/1.1 200
<?xml version="1.0" encoding="UTF-8"?>
<PublicAccessBlockConfiguration>
  <BlockPublicAcls>boolean</BlockPublicAcls>
  <IgnorePublicAcls>boolean</IgnorePublicAcls>
  <BlockPublicPolicy>boolean</BlockPublicPolicy>
  <RestrictPublicBuckets>boolean</RestrictPublicBuckets>
</PublicAccessBlockConfiguration>

Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in XML format by the service.

PublicAccessBlockConfiguration (p. 464)

Root level tag for the PublicAccessBlockConfiguration parameters.
Required: Yes
BlockPublicAcls (p. 464)

Specifies whether Amazon S3 should block public access control lists (ACLs) for buckets in this account. Setting this element to TRUE causes the following behavior:

- PUT Bucket acl and PUT Object acl calls fail if the specified ACL is public.
- PUT Object calls fail if the request includes a public ACL.
- PUT Bucket calls fail if the request includes a public ACL.

Enabling this setting doesn't affect existing policies or ACLs.

This is not supported for Amazon S3 on Outposts.

Type: Boolean

BlockPublicPolicy (p. 464)

Specifies whether Amazon S3 should block public bucket policies for buckets in this account. Setting this element to TRUE causes Amazon S3 to reject calls to PUT Bucket policy if the specified bucket policy allows public access.

Enabling this setting doesn't affect existing bucket policies.

This is not supported for Amazon S3 on Outposts.

Type: Boolean

IgnorePublicAcls (p. 464)

Specifies whether Amazon S3 should ignore public ACLs for buckets in this account. Setting this element to TRUE causes Amazon S3 to ignore all public ACLs on buckets in this account and any objects that they contain.

Enabling this setting doesn't affect the persistence of any existing ACLs and doesn't prevent new public ACLs from being set.

This is not supported for Amazon S3 on Outposts.

Type: Boolean

RestrictPublicBuckets (p. 464)

Specifies whether Amazon S3 should restrict public bucket policies for buckets in this account. Setting this element to TRUE restricts access to buckets with public policies to only AWS service principals and authorized users within this account.

Enabling this setting doesn't affect previously stored bucket policies, except that public and cross-account access within any public bucket policy, including non-public delegation to specific accounts, is blocked.

This is not supported for Amazon S3 on Outposts.

Type: Boolean

Amazon S3 throws this exception if you make a GetPublicAccessBlock request against an account that doesn't have a PublicAccessBlockConfiguration set.

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
• AWS SDK for .NET
• AWS SDK for C++
• AWS SDK for Go
• AWS SDK for Java
• AWS SDK for JavaScript
• AWS SDK for PHP V3
• AWS SDK for Python
• AWS SDK for Ruby V3
ListAccessPoints
Service: AWS S3 Control

Returns a list of the access points currently associated with the specified bucket. You can retrieve up to 1000 access points per call. If the specified bucket has more than 1,000 access points (or the number specified in maxResults, whichever is less), the response will include a continuation token that you can use to list the additional access points.

All Amazon S3 on Outposts REST API requests for this action require an additional parameter of x-amz-outpost-id to be passed with the request and an S3 on Outposts endpoint hostname prefix instead of s3-control. For an example of the request syntax for Amazon S3 on Outposts that uses the S3 on Outposts endpoint hostname prefix and the x-amz-outpost-id derived using the access point ARN, see the Examples section.

The following actions are related to ListAccessPoints:

- CreateAccessPoint
- DeleteAccessPoint
- GetAccessPoint

Request Syntax

GET /v20180820/accesspoint?bucket=Bucket&maxResults=MaxResults&nextToken=NextToken HTTP/1.1
x-amz-account-id: AccountId

URI Request Parameters

The request uses the following URI parameters.

**bucket (p. 467)**

The name of the bucket whose associated access points you want to list.

For using this parameter with Amazon S3 on Outposts with the REST API, you must specify the name and the x-amz-outpost-id as well.

For using this parameter with S3 on Outposts with the AWS SDK and CLI, you must specify the ARN of the bucket accessed in the format `arn:aws:s3-outposts:<Region>:<account-id>:outpost/<outpost-id>/bucket/<my-bucket-name>`. For example, to access the bucket reports through outpost my-outpost owned by account 123456789012 in Region us-west-2, use the URL encoding of `arn:aws:s3-outposts:us-west-2:123456789012:outpost/my-outpost/bucket/reports`. The value must be URL encoded.


**maxResults (p. 467)**

The maximum number of access points that you want to include in the list. If the specified bucket has more than this number of access points, then the response will include a continuation token in the NextToken field that you can use to retrieve the next page of access points.

Valid Range: Minimum value of 0. Maximum value of 1000.

**nextToken (p. 467)**

A continuation token. If a previous call to ListAccessPoints returned a continuation token in the NextToken field, then providing that value here causes Amazon S3 to retrieve the next page of results.

**x-amz-account-id (p. 467)**

The AWS account ID for owner of the bucket whose access points you want to list.

Length Constraints: Maximum length of 64.

Pattern: ^\d{12}$

Required: Yes

**Request Body**

The request does not have a request body.

**Response Syntax**

```xml
HTTP/1.1 200
<?xml version="1.0" encoding="UTF-8"?>
<ListAccessPointsResult>
  <AccessPointList>
    <AccessPoint>
      <AccessPointArn>string</AccessPointArn>
      <Bucket>string</Bucket>
      <Name>string</Name>
      <NetworkOrigin>string</NetworkOrigin>
      <VpcConfiguration>
        <VpcId>string</VpcId>
      </VpcConfiguration>
    </AccessPoint>
  </AccessPointList>
  <NextToken>string</NextToken>
</ListAccessPointsResult>
```

**Response Elements**

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in XML format by the service.

**ListAccessPointsResult (p. 468)**

Root level tag for the ListAccessPointsResult parameters.

Required: Yes

**AccessPointList (p. 468)**

Contains identification and configuration information for one or more access points associated with the specified bucket.

Type: Array of **AccessPoint (p. 679)** data types

**NextToken (p. 468)**

If the specified bucket has more access points than can be returned in one call to this API, this field contains a continuation token that you can provide in subsequent calls to this API to retrieve additional access points.

Type: String

Examples

Sample request syntax for ListAccessPoints for Amazon S3 on Outposts

The following request returns a list access points of the specified Amazon S3 on Outposts bucket example-outpost-bucket.

```
GET /v20180820/accesspoint?Bucket=example-outpost-bucket&MaxResults=MaxResults&NextToken=NextToken HTTP/1.1
Host: s3-outposts.<Region>.amazonaws.com
Date: Wed, 28 Oct 2020 22:32:00 GMT
Authorization: authorization string
x-amz-account-id: example-account-id
x-amz-outpost-id: op-01ac5d28a6a232904
```

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3
**ListJobs**

Service: AWS S3 Control

Lists current S3 Batch Operations jobs and jobs that have ended within the last 30 days for the AWS account making the request. For more information, see S3 Batch Operations in the *Amazon Simple Storage Service Developer Guide*.

Related actions include:

- CreateJob
- DescribeJob
- UpdateJobPriority
- UpdateJobStatus

**Request Syntax**

```
GET /v20180820/jobs?jobStatuses=JobStatuses&maxResults=MaxResults&nextToken=NextToken
HTTP/1.1
x-amz-account-id: AccountId
```

**URI Request Parameters**

The request uses the following URI parameters.

**jobStatuses (p. 470)**

The List Jobs request returns jobs that match the statuses listed in this element.

Valid Values: Active | Cancelled | Cancelling | Complete | Completing | Failed | Failing | New | Paused | Pausing | Preparing | Ready | Suspended

**maxResults (p. 470)**

The maximum number of jobs that Amazon S3 will include in the List Jobs response. If there are more jobs than this number, the response will include a pagination token in the NextToken field to enable you to retrieve the next page of results.

Valid Range: Minimum value of 0. Maximum value of 1000.

**nextToken (p. 470)**

A pagination token to request the next page of results. Use the token that Amazon S3 returned in the NextToken element of the ListJobsResult from the previous List Jobs request.


Pattern: `^[A-Za-z0-9\+\:\/\?\=\-\_]\+$`

**x-amz-account-id (p. 470)**

Length Constraints: Maximum length of 64.

Pattern: `^\d\{12\}$`

Required: Yes

**Request Body**

The request does not have a request body.
Response Syntax

HTTP/1.1 200
<?xml version="1.0" encoding="UTF-8"?>
<ListJobsResult>
  <NextToken>string</NextToken>
  <Jobs>
    <JobListDescriptor>
      <CreationTime>timestamp</CreationTime>
      <Description>string</Description>
      <JobId>string</JobId>
      <Operation>string</Operation>
      <Priority>integer</Priority>
      <ProgressSummary>
        <NumberOfTasksFailed>long</NumberOfTasksFailed>
        <NumberOfTasksSucceeded>long</NumberOfTasksSucceeded>
        <TotalNumberOfTasks>long</TotalNumberOfTasks>
      </ProgressSummary>
      <Status>string</Status>
      <TerminationDate>timestamp</TerminationDate>
    </JobListDescriptor>
  </Jobs>
</ListJobsResult>

Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in XML format by the service.

ListJobsResult (p. 471)
   Root level tag for the ListJobsResult parameters.
   Required: Yes

Jobs (p. 471)
   The list of current jobs and jobs that have ended within the last 30 days.
   Type: Array of JobListDescriptor (p. 686) data types

NextToken (p. 471)
   If the List Jobs request produced more than the maximum number of results, you can pass this
   value into a subsequent List Jobs request in order to retrieve the next page of results.
   Type: String
   Pattern: ^[A-Za-z0-9:\/:\/?\-_]+$
- AWS SDK for Java
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3
ListRegionalBuckets
Service: AWS S3 Control

Returns a list of all Outposts buckets in an Outposts that are owned by the authenticated sender of the request. For more information, see Using Amazon S3 on Outposts in the Amazon Simple Storage Service Developer Guide.

For an example of the request syntax for Amazon S3 on Outposts that uses the S3 on Outposts endpoint hostname prefix and x-amz-outpost-id in your API request, see the Examples section.

Request Syntax

GET /v20180820/bucket?maxResults=MaxResults&nextToken=NextToken HTTP/1.1
x-amz-account-id: AccountId
x-amz-outpost-id: OutpostId

URI Request Parameters

The request uses the following URI parameters.

maxResults (p. 473)
Valid Range: Minimum value of 0. Maximum value of 1000.

nextToken (p. 473)

x-amz-account-id (p. 473)
The AWS account ID of the Outposts bucket.
Length Constraints: Maximum length of 64.
Pattern: ^\d{12}$
Required: Yes

x-amz-outpost-id (p. 473)
The ID of the AWS Outposts.

Note
This is required by Amazon S3 on Outposts buckets.
Length Constraints: Minimum length of 1. Maximum length of 64.

Request Body

The request does not have a request body.

Response Syntax

HTTP/1.1 200
<?xml version="1.0" encoding="UTF-8"?>
<ListRegionalBucketsResult>
  <RegionalBucketList>
    <RegionalBucket>
      <Bucket>string</Bucket>
      <BucketArn>string</BucketArn>
  </RegionalBucket>
</ListRegionalBucketsResult>
Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in XML format by the service.

**ListRegionalBucketsResult (p. 473)**

Root level tag for the ListRegionalBucketsResult parameters.

- **Required:** Yes

**NextToken (p. 473)**

NextToken is sent when isTruncated is true, which means there are more buckets that can be listed. The next list requests to Amazon S3 can be continued with this NextToken. NextToken is obfuscated and is not a real key.

- **Type:** String
- **Length Constraints:** Minimum length of 1. Maximum length of 1024.

**RegionalBucketList (p. 473)**

- **Type:** Array of RegionalBucket (p. 708) data types

---

**Examples**

**Sample request to list an account’s Outposts buckets**

This request lists regional buckets.

```
GET /v20180820/bucket HTTP /1.1
Host:s3-outposts.us-west-2.amazonaws.com
Content-Length: 0
x-amz-outpost-id: op-01ac5d28a6a232904
x-amz-account-id: example-account-id
Date: Wed, 01 Mar 2006 12:00:00 GMT
Authorization: authorization string
```

---

**See Also**

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
• AWS SDK for JavaScript
• AWS SDK for PHP V3
• AWS SDK for Python
• AWS SDK for Ruby V3
PutAccessPointPolicy
Service: AWS S3 Control

Associates an access policy with the specified access point. Each access point can have only one policy, so a request made to this API replaces any existing policy associated with the specified access point.

All Amazon S3 on Outposts REST API requests for this action require an additional parameter of x-amz-outpost-id to be passed with the request and an S3 on Outposts endpoint hostname prefix instead of s3-control. For an example of the request syntax for Amazon S3 on Outposts that uses the S3 on Outposts endpoint hostname prefix and the x-amz-outpost-id derived using the access point ARN, see the Examples section.

The following actions are related to PutAccessPointPolicy:

- GetAccessPointPolicy
- DeleteAccessPointPolicy

Request Syntax

```
PUT /v20180820/accesspoint/name/policy HTTP/1.1
x-amz-account-id: AccountId
<?xml version="1.0" encoding="UTF-8"?>
  <Policy>string</Policy>
</PutAccessPointPolicyRequest>
```

URI Request Parameters

The request uses the following URI parameters.

**name (p. 476)**

The name of the access point that you want to associate with the specified policy.

For using this parameter with Amazon S3 on Outposts with the REST API, you must specify the name and the x-amz-outpost-id as well.

For using this parameter with S3 on Outposts with the AWS SDK and CLI, you must specify the ARN of the access point accessed in the format arn:aws:s3-outposts:<Region>:<account-id>:outpost/<outpost-id>/accesspoint/<my-accesspoint-name>. For example, to access the access point reports-ap through outpost my-outpost owned by account 123456789012 in Region us-west-2, use the URL encoding of arn:aws:s3-outposts:us-west-2:123456789012:outpost/my-outpost/accesspoint/reports-ap. The value must be URL encoded.


Required: Yes

**x-amz-account-id (p. 476)**

The AWS account ID for owner of the bucket associated with the specified access point.

Length Constraints: Maximum length of 64.

Pattern: ^\d{12}$

Required: Yes
Request Body

The request accepts the following data in XML format.

**PutAccessPointPolicyRequest (p. 476)**

Root level tag for the PutAccessPointPolicyRequest parameters.

Required: Yes

**Policy (p. 476)**

The policy that you want to apply to the specified access point. For more information about access point policies, see Managing Data Access with Amazon S3 Access Points in the Amazon Simple Storage Service Developer Guide.

Type: String

Required: Yes

Response Syntax

```
HTTP/1.1 200
```

Response Elements

If the action is successful, the service sends back an HTTP 200 response with an empty HTTP body.

Examples

Sample request syntax for the PutAccessPointPolicy action for Amazon S3 on Outposts access point

This example illustrates one usage of PutAccessPointPolicy.

```
PUT /v20180820/accesspoint/example-access-point/policy HTTP/1.1
Host: s3-outposts.<Region>.amazonaws.com
Date: Wed, 28 Oct 2020 22:32:00 GMT
Authorization: authorization string
x-amz-account-id: example-account-id
x-amz-outpost-id: op-01ac5d28a6a232904
<?xml version="1.0" encoding="UTF-8"?>

```

API Version 2006-03-01

477
}]
</Policy>
</PutAccessPointPolicyRequest>

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3
PutBucketLifecycleConfiguration
Service: AWS S3 Control

Note
This API action puts a lifecycle configuration to an Amazon S3 on Outposts bucket. To put a lifecycle configuration to an S3 bucket, see PutBucketLifecycleConfiguration in the Amazon Simple Storage Service API.

Creates a new lifecycle configuration for the Outposts bucket or replaces an existing lifecycle configuration. Outposts buckets only support lifecycle configurations that delete/expire objects after a certain period of time and abort incomplete multipart uploads. For more information, see Managing Lifecycle Permissions for Amazon S3 on Outposts.

All Amazon S3 on Outposts REST API requests for this action require an additional parameter of x-amz-outpost-id to be passed with the request and an S3 on Outposts endpoint hostname prefix instead of s3-control. For an example of the request syntax for Amazon S3 on Outposts that uses the S3 on Outposts endpoint hostname prefix and the x-amz-outpost-id derived using the access point ARN, see the Examples section.

The following actions are related to PutBucketLifecycleConfiguration:
- GetBucketLifecycleConfiguration
- DeleteBucketLifecycleConfiguration

Request Syntax

PUT /v20180820/bucket/name/lifecycleconfiguration HTTP/1.1
Host: Bucket.s3-control.amazonaws.com
x-amz-account-id: AccountId
<?xml version="1.0" encoding="UTF-8"?>
<LifecycleConfiguration xmlns="http://awss3control.amazonaws.com/doc/2018-08-20/">
  <Rules>
    <Rule>
      <AbortIncompleteMultipartUpload>
        <DaysAfterInitiation>integer</DaysAfterInitiation>
      </AbortIncompleteMultipartUpload>
      <Expiration>
        <Date>timestamp</Date>
        <Days>integer</Days>
        <ExpiredObjectDeleteMarker>boolean</ExpiredObjectDeleteMarker>
      </Expiration>
      <Filter>
        <And>
          <Prefix>string</Prefix>
          <Tags>
            <Tag>
              <Key>string</Key>
              <Value>string</Value>
            </Tag>
          </Tags>
        </And>
        <Prefix>string</Prefix>
        <Tag>
          <Key>string</Key>
          <Value>string</Value>
        </Tag>
      </Filter>
    </Rule>
  </Rules>
</LifecycleConfiguration>
<NoncurrentVersionTransitions>
   <NoncurrentVersionTransition>
      <NoncurrentDays>integer</NoncurrentDays>
      <StorageClass>string</StorageClass>
   </NoncurrentVersionTransition>
</NoncurrentVersionTransitions>
</Status>
<Transitions>
   <Transition>
      <Date>timestamp</Date>
      <Days>integer</Days>
      <StorageClass>string</StorageClass>
   </Transition>
</Transitions>
</Rules>
</LifecycleConfiguration>

**URI Request Parameters**

The request uses the following URI parameters.

**name (p. 479)**

The name of the bucket for which to set the configuration.


Required: Yes

**x-amz-account-id (p. 479)**

The AWS account ID of the Outposts bucket.

Length Constraints: Maximum length of 64.

Pattern: ^\d{12}$

Required: Yes

**Request Body**

The request accepts the following data in XML format.

**LifecycleConfiguration (p. 479)**

Root level tag for the LifecycleConfiguration parameters.

Required: Yes

**Rules (p. 479)**

A lifecycle rule for individual objects in an Outposts bucket.

Type: Array of LifecycleRule (p. 699) data types

Required: No

**Response Syntax**

HTTP/1.1 200

---

API Version 2006-03-01

480
Response Elements

If the action is successful, the service sends back an HTTP 200 response with an empty HTTP body.

Examples

Sample PutBucketLifecycleConfiguration request on an Amazon S3 on Outposts bucket

This request puts a lifecycle configuration on an Outposts bucket named example-outpost-bucket.

```plaintext
PUT /v20180820/bucket/example-outpost-bucket/lifecycleconfiguration HTTP/1.1
Host:s3-outposts.<Region>.amazonaws.com
x-amz-account-id: example-account-id
x-amz-outpost-id: op-01ac5d28a6a232904
Content-Length: 0
Date: Wed, 01 Mar 2006 12:00:00 GMT
Content-MD5: q6yJDlIkcBaGGfb3QLY69A==
Authorization: authorization string
Content-Length: 214

<LifecycleConfiguration>
  <Rule>
    <ID>id2</ID>
    <Filter>
      <Prefix>logs/</Prefix>
    </Filter>
    <Status>Enabled</Status>
    <Expiration>
      <Days>365</Days>
    </Expiration>
  </Rule>
</LifecycleConfiguration>
```

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3
**PutBucketPolicy**
Service: AWS S3 Control

**Note**
This API action puts a bucket policy to an Amazon S3 on Outposts bucket. To put a policy on an S3 bucket, see [PutBucketPolicy](#) in the [Amazon Simple Storage Service API](#).

Applies an Amazon S3 bucket policy to an Outposts bucket. For more information, see [Using Amazon S3 on Outposts](#) in the [Amazon Simple Storage Service Developer Guide](#).

If you are using an identity other than the root user of the AWS account that owns the Outposts bucket, the calling identity must have the `PutBucketPolicy` permissions on the specified Outposts bucket and belong to the bucket owner's account in order to use this operation.

If you don't have `PutBucketPolicy` permissions, Amazon S3 returns a 403 Access Denied error. If you have the correct permissions, but you're not using an identity that belongs to the bucket owner's account, Amazon S3 returns a 405 Method Not Allowed error.

**Important**
As a security precaution, the root user of the AWS account that owns a bucket can always use this operation, even if the policy explicitly denies the root user the ability to perform this action.

For more information about bucket policies, see [Using Bucket Policies and User Policies](#).

All Amazon S3 on Outposts REST API requests for this action require an additional parameter of `x-amz-outpost-id` to be passed with the request and an S3 on Outposts endpoint hostname prefix instead of `s3-control`. For an example of the request syntax for Amazon S3 on Outposts that uses the S3 on Outposts endpoint hostname prefix and the `x-amz-outpost-id` derived using the access point ARN, see the [Examples](#) section.

The following actions are related to `PutBucketPolicy`:

- GetBucketPolicy
- DeleteBucketPolicy

**Request Syntax**

PUT /v20180820/bucket/name/policy HTTP/1.1
Host: Bucket.s3-control.amazonaws.com
x-amz-account-id: AccountId
x-amz-confirm-remove-self-bucket-access: ConfirmRemoveSelfBucketAccess
<?xml version="1.0" encoding="UTF-8"?>
  <Policy>string</Policy>
</PutBucketPolicyRequest>

**URI Request Parameters**

The request uses the following URI parameters.

**name (p. 482)**

Specifies the bucket.

For using this parameter with Amazon S3 on Outposts with the REST API, you must specify the name and the `x-amz-outpost-id` as well.

For using this parameter with S3 on Outposts with the AWS SDK and CLI, you must specify the ARN of the bucket accessed in the format `arn:aws:s3-outposts:<Region>:<account-`
id>:outpost/<outpost-id>/bucket/<my-bucket-name>. For example, to access the bucket reports through outpost my-outpost owned by account 123456789012 in Region us-west-2, use the URL encoding of arn:aws:s3-outposts:us-west-2:123456789012:outpost/my-outpost/bucket/reports. The value must be URL encoded.


Required: Yes

x-amz-account-id (p. 482)

The AWS account ID of the Outposts bucket.

Length Constraints: Maximum length of 64.

Pattern: ^\d{12}$

Required: Yes

x-amz-confirm-remove-self-bucket-access (p. 482)

Set this parameter to true to confirm that you want to remove your permissions to change this bucket policy in the future.

Note
This is not supported by Amazon S3 on Outposts buckets.

Request Body

The request accepts the following data in XML format.

PutBucketPolicyRequest (p. 482)

Root level tag for the PutBucketPolicyRequest parameters.

Required: Yes

Policy (p. 482)

The bucket policy as a JSON document.

Type: String

Required: Yes

Response Syntax

HTTP/1.1 200

Response Elements

If the action is successful, the service sends back an HTTP 200 response with an empty HTTP body.

Examples

Sample request for putting a bucket policy in an Amazon S3 on Outposts bucket

The following request shows the PUT an individual policy request for the Outposts bucket example-outpost-bucket.
PUT v20180820/bucket/example-outpost-bucket/policy HTTP/1.1
Host: s3-outposts.<Region>.amazonaws.com
Date: Tue, 04 Apr 2010 20:34:56 GMT
Authorization: authorization string
x-amz-account-id: example-account-id
x-amz-outpost-id: op-01ac5d28a6a232904
{
    "Version":"2012-10-17",
    "Id":"exampleS3OnOutpostBucketPolicy",
    "Statement": [
        {
            "Sid":"st1",
            "Effect":"Allow",
            "Principal":{
                "AWS":"example-account-id"
            },
            "Action":"s3-outposts:*",
            "Resource":"arn:aws:s3-outposts:<your-region>:example-account-id:outpost/op-01ac5d28a6a232904/bucket/example-outpost-bucket"
        }
    ]
}
PutBucketTagging
Service: AWS S3 Control

Note
This API action puts tags on an Amazon S3 on Outposts bucket. To put tags on an S3 bucket, see PutBucketTagging in the Amazon Simple Storage Service API.

Sets the tags for an Outposts bucket. For more information, see Using Amazon S3 on Outposts in the Amazon Simple Storage Service Developer Guide.

Use tags to organize your AWS bill to reflect your own cost structure. To do this, sign up to get your AWS account bill with tag key values included. Then, to see the cost of combined resources, organize your billing information according to resources with the same tag key values. For example, you can tag several resources with a specific application name, and then organize your billing information to see the total cost of that application across several services. For more information, see Cost Allocation and Tagging.

Note
Within a bucket, if you add a tag that has the same key as an existing tag, the new value overwrites the old value. For more information, see Using Cost Allocation in Amazon S3 Bucket Tags.

To use this operation, you must have permissions to perform the s3-outposts:PutBucketTagging action. The Outposts bucket owner has this permission by default and can grant this permission to others. For more information about permissions, see Permissions Related to Bucket Subresource Operations and Managing Access Permissions to Your Amazon S3 Resources.

PutBucketTagging has the following special errors:

- Error code: InvalidTagError
  - Description: The tag provided was not a valid tag. This error can occur if the tag did not pass input validation. For information about tag restrictions, see User-Defined Tag Restrictions and AWS-Generated Cost Allocation Tag Restrictions.
- Error code: MalformedXMLError
  - Description: The XML provided does not match the schema.
- Error code: OperationAbortedError
  - Description: A conflicting conditional operation is currently in progress against this resource. Try again.
- Error code: InternalError
  - Description: The service was unable to apply the provided tag to the bucket.

All Amazon S3 on Outposts REST API requests for this action require an additional parameter of x-amz-outpost-id to be passed with the request and an S3 on Outposts endpoint hostname prefix instead of s3-control. For an example of the request syntax for Amazon S3 on Outposts that uses the S3 on Outposts endpoint hostname prefix and the x-amz-outpost-id derived using the access point ARN, see the Examples section.

The following actions are related to PutBucketTagging:

- GetBucketTagging
- DeleteBucketTagging

Request Syntax

```
PUT /v20180820/bucket/name/tagging HTTP/1.1
Host: Bucket.s3-control.amazonaws.com
```
x-amz-account-id: AccountId

<?xml version="1.0" encoding="UTF-8"?>
<Tagging xmlns="http://awss3control.amazonaws.com/doc/2018-08-20/">
  <TagSet>
    <Tag>
      <Key>string</Key>
      <Value>string</Value>
    </Tag>
  </TagSet>
</Tagging>

URI Request Parameters

The request uses the following URI parameters.

name (p. 485)

The Amazon Resource Name (ARN) of the bucket.

For using this parameter with Amazon S3 on Outposts with the REST API, you must specify the name and the x-amz-outpost-id as well.

For using this parameter with S3 on Outposts with the AWS SDK and CLI, you must specify the ARN of the bucket accessed in the format arn:aws:s3-outposts:<Region>:<account-id>:outpost/<outpost-id>/bucket/<my-bucket-name>. For example, to access the bucket reports through outpost my-outpost owned by account 123456789012 in Region us-west-2, use the URL encoding of arn:aws:s3-outposts:us-west-2:123456789012:outpost/my-outpost/bucket/reports. The value must be URL encoded.


Required: Yes

x-amz-account-id (p. 485)

The AWS account ID of the Outposts bucket.

Length Constraints: Maximum length of 64.

Pattern: ^\d{12}$

Required: Yes

Request Body

The request accepts the following data in XML format.

Tagging (p. 485)

Root level tag for the Tagging parameters.

Required: Yes

TagSet (p. 485)

A collection for a set of tags.

Type: Array of S3Tag (p. 726) data types

Required: Yes
Response Syntax

HTTP/1.1 200

Response Elements

If the action is successful, the service sends back an HTTP 200 response with an empty HTTP body.

Examples

Sample request: Add tag set to an Amazon S3 on Outposts bucket

The following request adds a tag set to the existing example-outpost-bucket bucket.

```xml
PUT v20180820/bucket/example-outpost-bucket/tagging HTTP/1.1
Host: s3-outposts.<Region>.amazonaws.com
Content-Length: 1660
x-amz-date: Thu, 12 Nov 2020 20:04:21 GMT
x-amz-account-id: example-account-id
x-amz-outpost-id: op-01ac5d28a6a232904
Authorization: authorization string

<Tagging>
  <TagSet>
    <Tag>
      <Key>Project</Key>
      <Value>Project One</Value>
    </Tag>
    <Tag>
      <Key>User</Key>
      <Value>jsmith</Value>
    </Tag>
  </TagSet>
</Tagging>
```

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3
PutJobTagging
Service: AWS S3 Control

Sets the supplied tag-set on an S3 Batch Operations job.

A tag is a key-value pair. You can associate S3 Batch Operations tags with any job by sending a PUT request against the tagging subresource that is associated with the job. To modify the existing tag set, you can either replace the existing tag set entirely, or make changes within the existing tag set by retrieving the existing tag set using GetJobTagging, modify that tag set, and use this API action to replace the tag set with the one you modified. For more information, see Controlling access and labeling jobs using tags in the Amazon Simple Storage Service Developer Guide.

Note

- If you send this request with an empty tag set, Amazon S3 deletes the existing tag set on the Batch Operations job. If you use this method, you are charged for a Tier 1 Request (PUT). For more information, see Amazon S3 pricing.
- For deleting existing tags for your Batch Operations job, a DeleteJobTagging request is preferred because it achieves the same result without incurring charges.
- A few things to consider about using tags:
  - Amazon S3 limits the maximum number of tags to 50 tags per job.
  - You can associate up to 50 tags with a job as long as they have unique tag keys.
  - A tag key can be up to 128 Unicode characters in length, and tag values can be up to 256 Unicode characters in length.
  - The key and values are case sensitive.
  - For tagging-related restrictions related to characters and encodings, see User-Defined Tag Restrictions in the AWS Billing and Cost Management User Guide.

To use this operation, you must have permission to perform the s3:PutJobTagging action.

Related actions include:

- CreateJob
- GetJobTagging
- DeleteJobTagging

Request Syntax

```xml
PUT /v20180820/jobs/id/tagging HTTP/1.1
x-amz-account-id: AccountId
<?xml version="1.0" encoding="UTF-8"?>
<PutJobTaggingRequest xmlns="http://awss3control.amazonaws.com/doc/2018-08-20/">
  <Tags>
    <Tag>
      <Key>string</Key>
      <Value>string</Value>
    </Tag>
  </Tags>
</PutJobTaggingRequest>
```

URI Request Parameters

The request uses the following URI parameters.

API Version 2006-03-01
488
id (p. 488)

The ID for the S3 Batch Operations job whose tags you want to replace.


Pattern: [a-zA-Z0-9\-\_]+

Required: Yes

x-amz-account-id (p. 488)

The AWS account ID associated with the S3 Batch Operations job.

Length Constraints: Maximum length of 64.

Pattern: ^\d{12}$

Required: Yes

Request Body

The request accepts the following data in XML format.

PutJobTaggingRequest (p. 488)

Root level tag for the PutJobTaggingRequest parameters.

Required: Yes

Tags (p. 488)

The set of tags to associate with the S3 Batch Operations job.

Type: Array of S3Tag (p. 726) data types

Required: Yes

Response Syntax

```
HTTP/1.1 200
```

Response Elements

If the action is successful, the service sends back an HTTP 200 response with an empty HTTP body. Amazon S3 throws this exception if you have too many tags in your tag set.

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for JavaScript
• AWS SDK for PHP V3
• AWS SDK for Python
• AWS SDK for Ruby V3
PutPublicAccessBlock

Service: AWS S3 Control

Creates or modifies the PublicAccessBlock configuration for an AWS account. For more information, see Using Amazon S3 block public access.

Related actions include:
- GetPublicAccessBlock
- DeletePublicAccessBlock

Request Syntax

```xml
PUT /v20180820/configuration/publicAccessBlock HTTP/1.1
x-amz-account-id: AccountId
<?xml version="1.0" encoding="UTF-8"?>
<PublicAccessBlockConfiguration xmlns="http://awss3control.amazonaws.com/doc/2018-08-20/">
  <BlockPublicAcls>boolean</BlockPublicAcls>
  <IgnorePublicAcls>boolean</IgnorePublicAcls>
  <BlockPublicPolicy>boolean</BlockPublicPolicy>
  <RestrictPublicBuckets>boolean</RestrictPublicBuckets>
</PublicAccessBlockConfiguration>
```

URI Request Parameters

The request uses the following URI parameters.

**x-amz-account-id (p. 491)**

The account ID for the AWS account whose PublicAccessBlock configuration you want to set.

Length Constraints: Maximum length of 64.

Pattern: `^\d{12}$`

Required: Yes

Request Body

The request accepts the following data in XML format.

**PublicAccessBlockConfiguration (p. 491)**

Root level tag for the PublicAccessBlockConfiguration parameters.

Required: Yes

**BlockPublicAcls (p. 491)**

Specifies whether Amazon S3 should block public access control lists (ACLs) for buckets in this account. Setting this element to TRUE causes the following behavior:

- PUT Bucket acl and PUT Object acl calls fail if the specified ACL is public.
- PUT Object calls fail if the request includes a public ACL.
- PUT Bucket calls fail if the request includes a public ACL.

Enabling this setting doesn't affect existing policies or ACLs.

This is not supported for Amazon S3 on Outposts.
Type: Boolean
Required: No
BlockPublicPolicy (p. 491)
Specifies whether Amazon S3 should block public bucket policies for buckets in this account. Setting this element to TRUE causes Amazon S3 to reject calls to PUT Bucket policy if the specified bucket policy allows public access.

Enabling this setting doesn't affect existing bucket policies.

This is not supported for Amazon S3 on Outposts.

Type: Boolean
Required: No
IgnorePublicAcls (p. 491)
Specifies whether Amazon S3 should ignore public ACLs for buckets in this account. Setting this element to TRUE causes Amazon S3 to ignore all public ACLs on buckets in this account and any objects that they contain.

Enabling this setting doesn't affect the persistence of any existing ACLs and doesn't prevent new public ACLs from being set.

This is not supported for Amazon S3 on Outposts.

Type: Boolean
Required: No
RestrictPublicBuckets (p. 491)
Specifies whether Amazon S3 should restrict public bucket policies for buckets in this account. Setting this element to TRUE restricts access to buckets with public policies to only AWS service principals and authorized users within this account.

Enabling this setting doesn't affect previously stored bucket policies, except that public and cross-account access within any public bucket policy, including non-public delegation to specific accounts, is blocked.

This is not supported for Amazon S3 on Outposts.

Type: Boolean
Required: No

Response Syntax

HTTP/1.1 200

Response Elements
If the action is successful, the service sends back an HTTP 200 response with an empty HTTP body.

See Also
For more information about using this API in one of the language-specific AWS SDKs, see the following:
• AWS Command Line Interface
• AWS SDK for .NET
• AWS SDK for C++
• AWS SDK for Go
• AWS SDK for Java
• AWS SDK for JavaScript
• AWS SDK for PHP V3
• AWS SDK for Python
• AWS SDK for Ruby V3
UpdateJobPriority
Service: AWS S3 Control

Updates an existing S3 Batch Operations job's priority. For more information, see S3 Batch Operations in the Amazon Simple Storage Service Developer Guide.

Related actions include:
- CreateJob
- ListJobs
- DescribeJob
- UpdateJobStatus

Request Syntax

POST /v20180820/jobs/{id}/priority?priority={Priority} HTTP/1.1
x-amz-account-id: {AccountId}

URI Request Parameters

The request uses the following URI parameters.

id (p. 494)

The ID for the job whose priority you want to update.


Pattern: [a-zA-Z0-9\-\_]+

Required: Yes

priority (p. 494)

The priority you want to assign to this job.

Valid Range: Minimum value of 0. Maximum value of 2147483647.

Required: Yes

x-amz-account-id (p. 494)

Length Constraints: Maximum length of 64.

Pattern: ^\d\12$*

Required: Yes

Request Body

The request does not have a request body.

Response Syntax

HTTP/1.1 200
<?xml version="1.0" encoding="UTF-8"?>
<UpdateJobPriorityResult>
Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in XML format by the service.

**UpdateJobPriorityResult (p. 494)**

Root level tag for the UpdateJobPriorityResult parameters.

Required: Yes

**JobId (p. 494)**

The ID for the job whose priority Amazon S3 updated.

Type: String


Pattern: \[a-zA-Z0-9\-_\]+

**Priority (p. 494)**

The new priority assigned to the specified job.

Type: Integer

Valid Range: Minimum value of 0. Maximum value of 2147483647.

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3
**UpdateJobStatus**  
Service: AWS S3 Control

Updates the status for the specified job. Use this operation to confirm that you want to run a job or to cancel an existing job. For more information, see S3 Batch Operations in the Amazon Simple Storage Service Developer Guide.

Related actions include:

- CreateJob
- ListJobs
- DescribeJob
- UpdateJobStatus

**Request Syntax**

```plaintext
POST /v20180820/jobs/{id}/status?requestedJobStatus={requestedJobStatus}&statusUpdateReason={statusUpdateReason} HTTP/1.1
x-amz-account-id: AccountId
```

**URI Request Parameters**

The request uses the following URI parameters.

**id (p. 496)**

The ID of the job whose status you want to update.

- Pattern: `[a-zA-Z0-9\-_]+`
- Required: Yes

**requestedJobStatus (p. 496)**

The status that you want to move the specified job to.

- Valid Values: Cancelled | Ready
- Required: Yes

**statusUpdateReason (p. 496)**

A description of the reason why you want to change the specified job's status. This field can be any string up to the maximum length.


**x-amz-account-id (p. 496)**

- Length Constraints: Maximum length of 64.
- Pattern: `^[\d]{12}$`
- Required: Yes

**Request Body**

The request does not have a request body.
Response Syntax

HTTP/1.1 200
<?xml version="1.0" encoding="UTF-8"?>
<UpdateJobStatusResult>
  <JobId>string</JobId>
  <Status>string</Status>
  <StatusUpdateReason>string</StatusUpdateReason>
</UpdateJobStatusResult>

Response Elements

If the action is successful, the service sends back an HTTP 200 response. The following data is returned in XML format by the service.

UpdateJobStatusResult (p. 497)

Root level tag for the UpdateJobStatusResult parameters.
Required: Yes

JobId (p. 497)

The ID for the job whose status was updated.
Type: String
Pattern: [a-zA-Z0-9\-\_]+

Status (p. 497)

The current status for the specified job.
Type: String

Valid Values: Active | Cancelled | Cancelling | Complete | Completing | Failed |
| Failing | New | Paused | Pausing | Preparing | Ready | Suspended

StatusUpdateReason (p. 497)

The reason that the specified job's status was updated.
Type: String
Length Constraints: Minimum length of 1. Maximum length of 256.

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for JavaScript
- AWS SDK for PHP V3
• AWS SDK for Python
• AWS SDK for Ruby V3

Amazon S3 on Outposts

The following actions are supported by Amazon S3 on Outposts:

• CreateEndpoint (p. 499)
• DeleteEndpoint (p. 502)
• ListEndpoints (p. 504)
**CreateEndpoint**

Service: Amazon S3 on Outposts

Amazon S3 on Outposts access points simplify managing data access at scale for shared datasets in S3 on Outposts. S3 on Outposts uses endpoints to connect to Outposts buckets so that you can perform actions within your virtual private cloud (VPC). For more information, see Using Amazon S3 on Outposts in the Amazon Simple Storage Service Developer Guide.

This action creates an endpoint and associates it with the specified Outposts.

**Note**

It can take up to 20 minutes for this action to complete.

Related actions include:

- **DeleteEndpoint**
- **ListEndpoints**

**Request Syntax**

```
POST /S3Outposts/CreateEndpoint HTTP/1.1
Content-type: application/json
{
  "OutpostId": "string",
  "SecurityGroupId": "string",
  "SubnetId": "string"
}
```

**URI Request Parameters**

The request does not use any URI parameters.

**Request Body**

The request accepts the following data in JSON format.

**OutpostId (p. 499)**

The ID of the AWS Outposts.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 100.

Pattern: ^(op-[a-f0-9]{17}|\d{12}|ec2)$

Required: Yes

**SecurityGroupId (p. 499)**

The ID of the security group to use with the endpoint.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 100.

Pattern: ^sg-\d{8}|[0-9a-f]{17}$

Required: Yes
SubnetId (p. 499)
The ID of the subnet in the selected VPC.
Type: String
Length Constraints: Minimum length of 1. Maximum length of 100.
Pattern: ^subnet-(\[0-9a-f\]{8}|\[0-9a-f\]{17})$
Required: Yes

Response Syntax

HTTP/1.1 200
Content-type: application/json
{
   "EndpointArn": "string"
}

Response Elements
If the action is successful, the service sends back an HTTP 200 response.
The following data is returned in JSON format by the service.

EndpointArn (p. 500)
The Amazon Resource Name (ARN) of the endpoint.
Type: String
Pattern: ^arn:\(aws|aws-cn|aws-us-gov\|aws-iso|aws-iso-b\):s3-outposts:[a-zA-Z0-9\-0-9]*:【0-9\{12\}:outpost\/(op-【0-9\{17\}|ec2)/endpoint/【a-zA-Z0-9\{19\}$

Errors

AccessDeniedException
Access was denied for this action.
HTTP Status Code: 403

ConflictException
There was a conflict with this action, and it could not be completed.
HTTP Status Code: 409

InternalServerException
There was an exception with the internal server.
HTTP Status Code: 500

ResourceNotFoundException
The requested resource was not found.
HTTP Status Code: 404

**ValidationException**

There was an exception validating this data.

HTTP Status Code: 400

**See Also**

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3
DeleteEndpoint
Service: Amazon S3 on Outposts

Amazon S3 on Outposts access points simplify managing data access at scale for shared datasets in S3 on Outposts. S3 on Outposts uses endpoints to connect to Outposts buckets so that you can perform actions within your virtual private cloud (VPC). For more information, see Using Amazon S3 on Outposts in the Amazon Simple Storage Service Developer Guide.

This action deletes an endpoint.

**Note**
It can take up to 20 minutes for this action to complete.

Related actions include:
- CreateEndpoint
- ListEndpoints

**Request Syntax**

```
DELETE /S3Outposts/DeleteEndpoint?endpointId=EndpointId&outpostId=OutpostId HTTP/1.1
```

**URI Request Parameters**

The request uses the following URI parameters.

**EndpointId (p. 502)**

The ID of the endpoint.


Pattern: ^[a-zA-Z0-9]{19}$

Required: Yes

**OutpostId (p. 502)**

The ID of the AWS Outposts.

Length Constraints: Minimum length of 1. Maximum length of 100.

Pattern: ^(op-[a-f0-9]{17}|\d{12}|ec2)$

Required: Yes

**Request Body**

The request does not have a request body.

**Response Syntax**

```
HTTP/1.1 200
```

**Response Elements**

If the action is successful, the service sends back an HTTP 200 response with an empty HTTP body.
Errors

AccessDeniedException
Access was denied for this action.
HTTP Status Code: 403

InternalServerException
There was an exception with the internal server.
HTTP Status Code: 500

ResourceNotFoundException
The requested resource was not found.
HTTP Status Code: 404

ValidationException
There was an exception validating this data.
HTTP Status Code: 400

See Also
For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3
ListEndpoints
Service: Amazon S3 on Outposts

Amazon S3 on Outposts access points simplify managing data access at scale for shared datasets in S3 on Outposts. S3 on Outposts uses endpoints to connect to Outposts buckets so that you can perform actions within your virtual private cloud (VPC). For more information, see Using Amazon S3 on Outposts in the Amazon Simple Storage Service Developer Guide.

This action lists endpoints associated with the Outposts.

Related actions include:

- CreateEndpoint
- DeleteEndpoint

Request Syntax

```
GET /S3Outposts/ListEndpoints?maxResults=MaxResults&nextToken=NextToken HTTP/1.1
```

URI Request Parameters

The request uses the following URI parameters.

MaxResults (p. 504)

The max number of endpoints that can be returned on the request.

Valid Range: Minimum value of 0. Maximum value of 100.

NextToken (p. 504)

The next endpoint requested in the list.


Pattern: ^[A-Za-z0-9\+\:/=\?\#-\_]+$

Request Body

The request does not have a request body.

Response Syntax

```
HTTP/1.1 200
Content-type: application/json
{
  "Endpoints": [
    {
      "CidrBlock": "string",
      "CreationTime": number,
      "EndpointArn": "string",
      "NetworkInterfaces": [
        {
          "NetworkInterfaceId": "string"
        }
      ],
      "OutpostsId": "string",
    }
  
```
Response Elements

If the action is successful, the service sends back an HTTP 200 response. The following data is returned in JSON format by the service.

Endpoints (p. 504)

Returns an array of endpoints associated with AWS Outposts.

Type: Array of Endpoint (p. 730) objects

NextToken (p. 504)

The next endpoint returned in the list.

Type: String


Pattern: ^[A-Za-z0-9\+\:\/\?\=\-\_]\+$

Errors

AccessDeniedException

Access was denied for this action.

HTTP Status Code: 403

InternalServerErrorException

There was an exception with the internal server.

HTTP Status Code: 500

ResourceNotFoundException

The requested resource was not found.

HTTP Status Code: 404

ValidationException

There was an exception validating this data.

HTTP Status Code: 400

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
Data Types

The following data types are supported by Amazon Simple Storage Service:

- AbortIncompleteMultipartUpload (p. 514)
- AccelerateConfiguration (p. 515)
- AccessControlPolicy (p. 516)
- AccessControlTranslation (p. 517)
- AnalyticsAndOperator (p. 518)
- AnalyticsConfiguration (p. 519)
- AnalyticsExportDestination (p. 520)
- AnalyticsFilter (p. 521)
- AnalyticsS3BucketDestination (p. 522)
- Bucket (p. 523)
- BucketLifecycleConfiguration (p. 524)
- BucketLoggingStatus (p. 525)
- CloudFunctionConfiguration (p. 526)
- CommonPrefix (p. 528)
- CompletedMultipartUpload (p. 529)
- CompletedPart (p. 530)
- Condition (p. 531)
- ContinuationEvent (p. 532)
- CopyObjectResult (p. 533)
- CopyPartResult (p. 534)
- CORSConfiguration (p. 535)
- CORSRule (p. 536)
- CreateBucketConfiguration (p. 537)
- CSVInput (p. 538)
- CSVOutput (p. 540)
- DefaultRetention (p. 542)
- Delete (p. 543)
- DeletedObject (p. 544)
- DeleteMarkerEntry (p. 545)
- DeleteMarkerReplication (p. 546)
- Destination (p. 547)
- Encryption (p. 549)
- EncryptionConfiguration (p. 550)
- EndEvent (p. 551)
• Error (p. 552)
• ErrorDocument (p. 562)
• ExistingObjectReplication (p. 563)
• FilterRule (p. 564)
• GlacierJobParameters (p. 565)
• Grant (p. 566)
• Grantee (p. 567)
• IndexDocument (p. 569)
• Initiator (p. 570)
• InputSerialization (p. 571)
• IntelligentTieringAndOperator (p. 572)
• IntelligentTieringConfiguration (p. 573)
• IntelligentTieringFilter (p. 574)
• InventoryConfiguration (p. 575)
• InventoryDestination (p. 577)
• InventoryEncryption (p. 578)
• InventoryFilter (p. 579)
• InventoryS3BucketDestination (p. 580)
• InventorySchedule (p. 582)
• JSONInput (p. 583)
• JSONOutput (p. 584)
• LambdaFunctionConfiguration (p. 585)
• LifecycleConfiguration (p. 587)
• LifecycleExpiration (p. 588)
• LifecycleRule (p. 589)
• LifecycleRuleAndOperator (p. 591)
• LifecycleRuleFilter (p. 592)
• LoggingEnabled (p. 593)
• MetadataEntry (p. 594)
• Metrics (p. 595)
• MetricsAndOperator (p. 596)
• MetricsConfiguration (p. 597)
• MetricsFilter (p. 598)
• MultipartUpload (p. 599)
• NoncurrentVersionExpiration (p. 601)
• NoncurrentVersionTransition (p. 602)
• NotificationConfiguration (p. 603)
• NotificationConfigurationDeprecated (p. 604)
• NotificationConfigurationFilter (p. 605)
• Object (p. 606)
• ObjectIdentifier (p. 608)
• ObjectLockConfiguration (p. 609)
• ObjectLockLegalHold (p. 610)
• ObjectLockRetention (p. 611)
• ObjectLockRule (p. 612)
• ObjectVersion (p. 613)
• OutputLocation (p. 615)
• OutputSerialization (p. 616)
• Owner (p. 617)
• OwnershipControls (p. 618)
• OwnershipControlsRule (p. 619)
• ParquetInput (p. 620)
• Part (p. 621)
• PolicyStatus (p. 622)
• Progress (p. 623)
• ProgressEvent (p. 624)
• PublicAccessBlockConfiguration (p. 625)
• QueueConfiguration (p. 627)
• QueueConfigurationDeprecated (p. 629)
• RecordsEvent (p. 631)
• Redirect (p. 632)
• RedirectAllRequestsTo (p. 634)
• ReplicationConfiguration (p. 635)
• ReplicationRule (p. 636)
• ReplicationRuleAndOperator (p. 638)
• ReplicationRuleFilter (p. 639)
• ReplicationTime (p. 640)
• ReplicationTimeValue (p. 641)
• RequestPaymentConfiguration (p. 642)
• RequestProgress (p. 643)
• RestoreRequest (p. 644)
• RoutingRule (p. 646)
• Rule (p. 647)
• S3KeyFilter (p. 649)
• S3Location (p. 650)
• ScanRange (p. 652)
• SelectObjectContentEventStream (p. 653)
• SelectParameters (p. 654)
• ServerSideEncryptionByDefault (p. 655)
• ServerSideEncryptionConfiguration (p. 656)
• ServerSideEncryptionRule (p. 657)
• SourceSelectionCriteria (p. 658)
• SSEKMS (p. 659)
• SseKmsEncryptedObjects (p. 660)
• SSES3 (p. 661)
• Stats (p. 662)
• StatsEvent (p. 663)
• StorageClassAnalysis (p. 664)
• StorageClassAnalysisDataExport (p. 665)
• Tag (p. 666)
• Tagging (p. 667)
• TargetGrant (p. 668)
The following data types are supported by AWS S3 Control:

- AbortIncompleteMultipartUpload (p. 678)
- AccessPoint (p. 679)
- CreateBucketConfiguration (p. 681)
- JobDescriptor (p. 682)
- JobFailure (p. 685)
- JobListDescriptor (p. 686)
- JobOperation (p. 691)
- JobProgressSummary (p. 693)
- JobReport (p. 694)
- LambdaInvokeOperation (p. 696)
- LifecycleConfiguration (p. 697)
- LifecycleExpiration (p. 698)
- LifecycleRule (p. 699)
- LifecycleRuleAndOperator (p. 701)
- LifecycleRuleFilter (p. 702)
- NoncurrentVersionExpiration (p. 703)
- NoncurrentVersionTransition (p. 704)
- PolicyStatus (p. 705)
- PublicAccessBlockConfiguration (p. 706)
- RegionalBucket (p. 708)
- S3AccessControlList (p. 709)
- S3AccessControlPolicy (p. 710)
- S3CopyObjectOperation (p. 711)
- S3Grant (p. 714)
- S3Grantee (p. 715)
- S3InitiateRestoreObjectOperation (p. 716)
- S3ObjectLockLegalHold (p. 717)
- S3ObjectMetadata (p. 718)
- S3ObjectOwner (p. 720)
- S3Retention (p. 721)
- S3SetObjectAclOperation (p. 722)
- S3SetObjectLegalHoldOperation (p. 723)
- S3SetObjectRetentionOperation (p. 724)
- S3SetObjectTaggingOperation (p. 725)
The following data types are supported by Amazon S3 on Outposts:

- Endpoint (p. 730)
- NetworkInterface (p. 732)

Amazon Simple Storage Service

The following data types are supported by Amazon Simple Storage Service:

- AbortIncompleteMultipartUpload (p. 514)
- AccelerateConfiguration (p. 515)
- AccessControlPolicy (p. 516)
- AccessControlTranslation (p. 517)
- AnalyticsAndOperator (p. 518)
- AnalyticsConfiguration (p. 519)
- AnalyticsExportDestination (p. 520)
- AnalyticsFilter (p. 521)
- AnalyticsS3BucketDestination (p. 522)
- Bucket (p. 523)
- BucketLifecycleConfiguration (p. 524)
- BucketLoggingStatus (p. 525)
- CloudFunctionConfiguration (p. 526)
- CommonPrefix (p. 528)
- CompletedMultipartUpload (p. 529)
- CompletedPart (p. 530)
- Condition (p. 531)
- ContinuationEvent (p. 532)
- CopyObjectResult (p. 533)
- CopyPartResult (p. 534)
- CORSConfiguration (p. 535)
- CORSRule (p. 536)
- CreateBucketConfiguration (p. 537)
- CSVInput (p. 538)
- CSVOOutput (p. 540)
- DefaultRetention (p. 542)
- Delete (p. 543)
- DeletedObject (p. 544)
- DeleteMarkerEntry (p. 545)
- DeleteMarkerReplication (p. 546)
- Destination (p. 547)
- Encryption (p. 549)
Amazon Simple Storage Service API Reference
Amazon Simple Storage Service

- EncryptionConfiguration (p. 550)
- EndEvent (p. 551)
- Error (p. 552)
- ErrorDocument (p. 562)
- ExistingObjectReplication (p. 563)
- FilterRule (p. 564)
- GlacierJobParameters (p. 565)
- Grant (p. 566)
- Grantee (p. 567)
- IndexDocument (p. 569)
- Initiator (p. 570)
- InputSerialization (p. 571)
- IntelligentTieringAndOperator (p. 572)
- IntelligentTieringConfiguration (p. 573)
- IntelligentTieringFilter (p. 574)
- InventoryConfiguration (p. 575)
- InventoryDestination (p. 577)
- InventoryEncryption (p. 578)
- InventoryFilter (p. 579)
- InventoryS3BucketDestination (p. 580)
- InventorySchedule (p. 582)
- JSONInput (p. 583)
- JSONOutput (p. 584)
- LambdaFunctionConfiguration (p. 585)
- LifecycleConfiguration (p. 587)
- LifecycleExpiration (p. 588)
- LifecycleRule (p. 589)
- LifecycleRuleAndOperator (p. 591)
- LifecycleRuleFilter (p. 592)
- LoggingEnabled (p. 593)
- MetadataEntry (p. 594)
- Metrics (p. 595)
- MetricsAndOperator (p. 596)
- MetricsConfiguration (p. 597)
- MetricsFilter (p. 598)
- MultipartUpload (p. 599)
- NoncurrentVersionExpiration (p. 601)
- NoncurrentVersionTransition (p. 602)
- NotificationConfiguration (p. 603)
- NotificationConfigurationDeprecated (p. 604)
- NotificationConfigurationFilter (p. 605)
- Object (p. 606)
- ObjectIdentifier (p. 608)
- ObjectLockConfiguration (p. 609)
- ObjectLockLegalHold (p. 610)
- ObjectLockRetention (p. 611)
• ObjectLockRule (p. 612)
• ObjectVersion (p. 613)
• OutputLocation (p. 615)
• OutputSerialization (p. 616)
• Owner (p. 617)
• OwnershipControls (p. 618)
• OwnershipControlsRule (p. 619)
• ParquetInput (p. 620)
• Part (p. 621)
• PolicyStatus (p. 622)
• Progress (p. 623)
• ProgressEvent (p. 624)
• PublicAccessBlockConfiguration (p. 625)
• QueueConfiguration (p. 627)
• QueueConfigurationDeprecated (p. 629)
• RecordsEvent (p. 631)
• Redirect (p. 632)
• RedirectAllRequestsTo (p. 634)
• ReplicationConfiguration (p. 635)
• ReplicationRule (p. 636)
• ReplicationRuleAndOperator (p. 638)
• ReplicationRuleFilter (p. 639)
• ReplicationTime (p. 640)
• ReplicationTimeValue (p. 641)
• RequestPaymentConfiguration (p. 642)
• RequestProgress (p. 643)
• RestoreRequest (p. 644)
• RoutingRule (p. 646)
• Rule (p. 647)
• S3KeyFilter (p. 649)
• S3Location (p. 650)
• ScanRange (p. 652)
• SelectObjectContentEventStream (p. 653)
• SelectParameters (p. 654)
• ServerSideEncryptionByDefault (p. 655)
• ServerSideEncryptionConfiguration (p. 656)
• ServerSideEncryptionRule (p. 657)
• SourceSelectionCriteria (p. 658)
• SSEKMS (p. 659)
• SseKmsEncryptedObjects (p. 660)
• SSES3 (p. 661)
• Stats (p. 662)
• StatsEvent (p. 663)
• StorageClassAnalysis (p. 664)
• StorageClassAnalysisDataExport (p. 665)
• Tag (p. 666)
- Tagging (p. 667)
- TargetGrant (p. 668)
- Tiering (p. 669)
- TopicConfiguration (p. 670)
- TopicConfigurationDeprecated (p. 672)
- Transition (p. 674)
- VersioningConfiguration (p. 675)
- WebsiteConfiguration (p. 676)
AbortIncompleteMultipartUpload
Service: Amazon Simple Storage Service

Specifies the days since the initiation of an incomplete multipart upload that Amazon S3 will wait before permanently removing all parts of the upload. For more information, see Aborting Incomplete Multipart Uploads Using a Bucket Lifecycle Policy in the Amazon Simple Storage Service Developer Guide.

Contents

DaysAfterInitiation

Specifies the number of days after which Amazon S3 aborts an incomplete multipart upload.

Type: Integer
Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for Ruby V3
AccelerateConfiguration
Service: Amazon Simple Storage Service

Configures the transfer acceleration state for an Amazon S3 bucket. For more information, see Amazon S3 Transfer Acceleration in the Amazon Simple Storage Service Developer Guide.

Contents

Status

  Specifies the transfer acceleration status of the bucket.

  Type: String

  Valid Values:  Enabled | Suspended

  Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for Ruby V3
AccessControlPolicy
Service: Amazon Simple Storage Service

Contains the elements that set the ACL permissions for an object per grantee.

Contents

Grants

A list of grants.

Type: Array of Grant (p. 566) data types

Required: No

Owner

Container for the bucket owner’s display name and ID.

Type: Owner (p. 617) data type

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for Ruby V3
AccessControlTranslation
Service: Amazon Simple Storage Service

A container for information about access control for replicas.

Contents

Owner

Specifies the replica ownership. For default and valid values, see PUT bucket replication in the Amazon Simple Storage Service API Reference.

Type: String

Valid Values: Destination

Required: Yes

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for Ruby V3
AnalyticsAndOperator
Service: Amazon Simple Storage Service

A conjunction (logical AND) of predicates, which is used in evaluating a metrics filter. The operator must have at least two predicates in any combination, and an object must match all of the predicates for the filter to apply.

Contents

Prefix

The prefix to use when evaluating an AND predicate: The prefix that an object must have to be included in the metrics results.

Type: String
Required: No

Tags

The list of tags to use when evaluating an AND predicate.

Type: Array of Tag (p. 666) data types
Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for Ruby V3
AnalyticsConfiguration
Service: Amazon Simple Storage Service

Specifies the configuration and any analyses for the analytics filter of an Amazon S3 bucket.

Contents

Filter

The filter used to describe a set of objects for analyses. A filter must have exactly one prefix, one tag, or one conjunction (AnalyticsAndOperator). If no filter is provided, all objects will be considered in any analysis.

Type: AnalyticsFilter (p. 521) data type

Required: No

Id

The ID that identifies the analytics configuration.

Type: String

Required: Yes

StorageClassAnalysis

Contains data related to access patterns to be collected and made available to analyze the tradeoffs between different storage classes.

Type: StorageClassAnalysis (p. 664) data type

Required: Yes

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for Ruby V3
AnalyticsExportDestination
Service: Amazon Simple Storage Service

Where to publish the analytics results.

Contents

S3BucketDestination

A destination signifying output to an S3 bucket.

Type: AnalyticsS3BucketDestination (p. 522) data type

Required: Yes

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for Ruby V3
AnalyticsFilter
Service: Amazon Simple Storage Service

The filter used to describe a set of objects for analyses. A filter must have exactly one prefix, one tag, or one conjunction (AnalyticsAndOperator). If no filter is provided, all objects will be considered in any analysis.

Contents

And

A conjunction (logical AND) of predicates, which is used in evaluating an analytics filter. The operator must have at least two predicates.

Type: AnalyticsAndOperator (p. 518) data type

Required: No

Prefix

The prefix to use when evaluating an analytics filter.

Type: String

Required: No

Tag

The tag to use when evaluating an analytics filter.

Type: Tag (p. 666) data type

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for Ruby V3
AnalyticsS3BucketDestination
Service: Amazon Simple Storage Service

Contains information about where to publish the analytics results.

Contents

Bucket

The Amazon Resource Name (ARN) of the bucket to which data is exported.

Type: String
Required: Yes

BucketAccountId

The account ID that owns the destination S3 bucket. If no account ID is provided, the owner is not validated before exporting data.

Note
Although this value is optional, we strongly recommend that you set it to help prevent problems if the destination bucket ownership changes.

Type: String
Required: No

Format

Specifies the file format used when exporting data to Amazon S3.

Type: String
Valid Values: CSV
Required: Yes

Prefix

The prefix to use when exporting data. The prefix is prepended to all results.

Type: String
Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for Ruby V3
Bucket
Service: Amazon Simple Storage Service

In terms of implementation, a Bucket is a resource. An Amazon S3 bucket name is globally unique, and the namespace is shared by all AWS accounts.

Contents

CreationDate

Date the bucket was created.

Type: Timestamp
Required: No

Name

The name of the bucket.

Type: String
Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for Ruby V3
BucketLifecycleConfiguration
Service: Amazon Simple Storage Service

Specifies the lifecycle configuration for objects in an Amazon S3 bucket. For more information, see Object Lifecycle Management in the Amazon Simple Storage Service Developer Guide.

Contents

Rules

A lifecycle rule for individual objects in an Amazon S3 bucket.

Type: Array of LifecycleRule (p. 589) data types

Required: Yes

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for Ruby V3
BucketLoggingStatus
Service: Amazon Simple Storage Service

Container for logging status information.

Contents

LoggingEnabled

Describes where logs are stored and the prefix that Amazon S3 assigns to all log object keys for a bucket. For more information, see PUT Bucket logging in the Amazon Simple Storage Service API Reference.

Type: LoggingEnabled (p. 593) data type

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for Ruby V3
CloudFunctionConfiguration
Service: Amazon Simple Storage Service

Container for specifying the AWS Lambda notification configuration.

Contents

CloudFunction

Lambda cloud function ARN that Amazon S3 can invoke when it detects events of the specified type.

Type: String
Required: No

Event

This member has been deprecated.

The bucket event for which to send notifications.

Type: String


Required: No

Events

Bucket events for which to send notifications.

Type: Array of strings


Required: No

Id

An optional unique identifier for configurations in a notification configuration. If you don't provide one, Amazon S3 will assign an ID.

Type: String
Required: No

InvocationRole

The role supporting the invocation of the Lambda function
Type: String

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for Ruby V3
CommonPrefix
Service: Amazon Simple Storage Service

Container for all (if there are any) keys between Prefix and the next occurrence of the string specified by a delimiter. CommonPrefixes lists keys that act like subdirectories in the directory specified by Prefix. For example, if the prefix is notes/ and the delimiter is a slash (/) as in notes/summer/july, the common prefix is notes/summer/.

Contents

Prefix

Container for the specified common prefix.

Type: String

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for Ruby V3
CompletedMultipartUpload
Service: Amazon Simple Storage Service

The container for the completed multipart upload details.

Contents

Parts

Array of CompletedPart data types.

Type: Array of CompletedPart (p. 530) data types

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for Ruby V3
**CompletedPart**  
Service: Amazon Simple Storage Service  
Details of the parts that were uploaded.

**Contents**

**ETag**

Entity tag returned when the part was uploaded.

Type: String  
Required: No

**PartNumber**

Part number that identifies the part. This is a positive integer between 1 and 10,000.

Type: Integer  
Required: No

**See Also**

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for Ruby V3
**Condition**

Service: Amazon Simple Storage Service

A container for describing a condition that must be met for the specified redirect to apply. For example, 1. If request is for pages in the /docs folder, redirect to the /documents folder. 2. If request results in HTTP error 4xx, redirect request to another host where you might process the error.

**Contents**

**HttpErrorCodeReturnedEquals**

The HTTP error code when the redirect is applied. In the event of an error, if the error code equals this value, then the specified redirect is applied. Required when parent element **Condition** is specified and sibling **KeyPrefixEquals** is not specified. If both are specified, then both must be true for the redirect to be applied.

Type: String

Required: No

**KeyPrefixEquals**

The object key name prefix when the redirect is applied. For example, to redirect requests for ExamplePage.html, the key prefix will be ExamplePage.html. To redirect request for all pages with the prefix docs/, the key prefix will be /docs, which identifies all objects in the docs/ folder. Required when the parent element **Condition** is specified and sibling **HttpErrorCodeReturnedEquals** is not specified. If both conditions are specified, both must be true for the redirect to be applied.

Type: String

Required: No

**See Also**

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for Ruby V3
ContinuationEvent
Service: Amazon Simple Storage Service

Contents

The members of this structure are context-dependent.

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for Ruby V3
CopyObjectResult
Service: Amazon Simple Storage Service

Container for all response elements.

Contents

ETag

Returns the ETag of the new object. The ETag reflects only changes to the contents of an object, not its metadata. The source and destination ETag is identical for a successfully copied object.

Type: String
Required: No

LastModified

Returns the date that the object was last modified.

Type: Timestamp
Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for Ruby V3
CopyPartResult
Service: Amazon Simple Storage Service

Container for all response elements.

Contents

ETag
  Entity tag of the object.
  Type: String
  Required: No

LastModified
  Date and time at which the object was uploaded.
  Type: Timestamp
  Required: No

See Also
For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for Ruby V3
CORSConfiguration
Service: Amazon Simple Storage Service

Describes the cross-origin access configuration for objects in an Amazon S3 bucket. For more information, see Enabling Cross-Origin Resource Sharing in the Amazon Simple Storage Service Developer Guide.

Contents

CORSRules

A set of origins and methods (cross-origin access that you want to allow). You can add up to 100 rules to the configuration.

Type: Array of CORSRule (p. 536) data types

Required: Yes

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for Ruby V3
CORSRule

Service: Amazon Simple Storage Service

Specifies a cross-origin access rule for an Amazon S3 bucket.

Contents

AllowedHeaders

Headers that are specified in the Access-Control-Request-Headers header. These headers are allowed in a preflight OPTIONS request. In response to any preflight OPTIONS request, Amazon S3 returns any requested headers that are allowed.

Type: Array of strings

Required: No

AllowedMethods

An HTTP method that you allow the origin to execute. Valid values are GET, PUT, HEAD, POST, and DELETE.

Type: Array of strings

Required: Yes

AllowedOrigins

One or more origins you want customers to be able to access the bucket from.

Type: Array of strings

Required: Yes

ExposeHeaders

One or more headers in the response that you want customers to be able to access from their applications (for example, from a JavaScript XMLHttpRequest object).

Type: Array of strings

Required: No

MaxAgeSeconds

The time in seconds that your browser is to cache the preflight response for the specified resource.

Type: Integer

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for Ruby V3
CreateBucketConfiguration
Service: Amazon Simple Storage Service

The configuration information for the bucket.

Contents

LocationConstraint

Specifies the Region where the bucket will be created. If you don't specify a Region, the bucket is created in the US East (N. Virginia) Region (us-east-1).

Type: String

Valid Values: af-south-1 | ap-east-1 | ap-northeast-1 | ap-northeast-2 | ap-northeast-3 | ap-south-1 | ap-southeast-1 | ap-southeast-2 | ca-central-1 | cn-north-1 | cn-northwest-1 | EU | eu-central-1 | eu-north-1 | eu-south-1 | eu-west-1 | eu-west-2 | eu-west-3 | me-south-1 | sa-east-1 | us-east-2 | us-gov-east-1 | us-gov-west-1 | us-west-1 | us-west-2

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for Ruby V3
CSVInput
Service: Amazon Simple Storage Service

Describes how an uncompressed comma-separated values (CSV)-formatted input object is formatted.

Contents

AllowQuotedRecordDelimiter

Specifies that CSV field values may contain quoted record delimiters and such records should be allowed. Default value is FALSE. Setting this value to TRUE may lower performance.

Type: Boolean
Required: No

Comments

A single character used to indicate that a row should be ignored when the character is present at the start of that row. You can specify any character to indicate a comment line.

Type: String
Required: No

FieldDelimiter

A single character used to separate individual fields in a record. You can specify an arbitrary delimiter.

Type: String
Required: No

FileHeaderInfo

Describes the first line of input. Valid values are:

- NONE: First line is not a header.
- IGNORE: First line is a header, but you can't use the header values to indicate the column in an expression. You can use column position (such as _1, _2, ...) to indicate the column (SELECT s._1 FROM OBJECT s).
- USE: First line is a header, and you can use the header value to identify a column in an expression (SELECT "name" FROM OBJECT).

Type: String
Valid Values: USE | IGNORE | NONE
Required: No

QuoteCharacter

A single character used for escaping when the field delimiter is part of the value. For example, if the value is a, b, Amazon S3 wraps this field value in quotation marks, as follows: " a , b ".

Type: String
Default: "
Ancestors: CSV
Type: String
Required: No

**QuoteEscapeCharacter**

A single character used for escaping the quotation mark character inside an already escaped value. For example, the value """"a , b """" is parsed as " a , b "".

Type: String

Required: No

**RecordDelimiter**

A single character used to separate individual records in the input. Instead of the default value, you can specify an arbitrary delimiter.

Type: String

Required: No

**See Also**

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for Ruby V3
CSVOutput
Service: Amazon Simple Storage Service

Describes how uncompressed comma-separated values (CSV)-formatted results are formatted.

Contents

FieldDelimiter
The value used to separate individual fields in a record. You can specify an arbitrary delimiter.

Type: String
Required: No

QuoteCharacter
A single character used for escaping when the field delimiter is part of the value. For example, if the value is a, b, Amazon S3 wraps this field value in quotation marks, as follows: "a, b".

Type: String
Required: No

QuoteEscapeCharacter
The single character used for escaping the quote character inside an already escaped value.

Type: String
Required: No

QuoteFields
Indicates whether to use quotation marks around output fields.

- ALWAYS: Always use quotation marks for output fields.
- ASNEEDED: Use quotation marks for output fields when needed.

Type: String
Valid Values: ALWAYS | ASNEEDED
Required: No

RecordDelimiter
A single character used to separate individual records in the output. Instead of the default value, you can specify an arbitrary delimiter.

Type: String
Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for Ruby V3

API Version 2006-03-01
540
DefaultRetention
Service: Amazon Simple Storage Service

The container element for specifying the default Object Lock retention settings for new objects placed in
the specified bucket.

Contents

Days

The number of days that you want to specify for the default retention period.

Type: Integer

Required: No

Mode

The default Object Lock retention mode you want to apply to new objects placed in the specified
bucket.

Type: String

Valid Values: GOVERNANCE | COMPLIANCE

Required: No

Years

The number of years that you want to specify for the default retention period.

Type: Integer

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for Ruby V3
Delete
Service: Amazon Simple Storage Service

Container for the objects to delete.

Contents

Objects
The objects to delete.

Type: Array of ObjectIdentifier (p. 608) data types

Required: Yes

Quiet
Element to enable quiet mode for the request. When you add this element, you must set its value to true.

Type: Boolean

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for Ruby V3
DeletedObject
Service: Amazon Simple Storage Service

Information about the deleted object.

Contents

DeleteMarker

Specifies whether the versioned object that was permanently deleted was (true) or was not (false) a delete marker. In a simple DELETE, this header indicates whether (true) or not (false) a delete marker was created.

Type: Boolean
Required: No

DeleteMarkerVersionId

The version ID of the delete marker created as a result of the DELETE operation. If you delete a specific object version, the value returned by this header is the version ID of the object version deleted.

Type: String
Required: No

Key

The name of the deleted object.

Type: String
Length Constraints: Minimum length of 1.
Required: No

VersionId

The version ID of the deleted object.

Type: String
Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for Ruby V3
DeleteMarkerEntry
Service: Amazon Simple Storage Service

Information about the delete marker.

Contents

IsLatest
Specifies whether the object is (true) or is not (false) the latest version of an object.
Type: Boolean
Required: No

Key
The object key.
Type: String
Length Constraints: Minimum length of 1.
Required: No

LastModified
Date and time the object was last modified.
Type: Timestamp
Required: No

Owner
The account that created the delete marker.
Type: Owner (p. 617) data type
Required: No

VersionId
Version ID of an object.
Type: String
Required: No

See Also
For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for Ruby V3
DeleteMarkerReplication
Service: Amazon Simple Storage Service

Specifies whether Amazon S3 replicates delete markers. If you specify a Filter in your replication configuration, you must also include a DeleteMarkerReplication element. If your Filter includes a Tag element, the DeleteMarkerReplication Status must be set to Disabled, because Amazon S3 does not support replicating delete markers for tag-based rules. For an example configuration, see Basic Rule Configuration.

For more information about delete marker replication, see Basic Rule Configuration.

Note
If you are using an earlier version of the replication configuration, Amazon S3 handles replication of delete markers differently. For more information, see Backward Compatibility.

Contents

Status

Indicates whether to replicate delete markers.

Note
Indicates whether to replicate delete markers.

Type: String

Valid Values: Enabled | Disabled

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for Ruby V3
Destination
Service: Amazon Simple Storage Service

Specifies information about where to publish analysis or configuration results for an Amazon S3 bucket and S3 Replication Time Control (S3 RTC).

Contents

**AccessControlTranslation**

Specify this only in a cross-account scenario (where source and destination bucket owners are not the same), and you want to change replica ownership to the AWS account that owns the destination bucket. If this is not specified in the replication configuration, the replicas are owned by same AWS account that owns the source object.

Type: AccessControlTranslation (p. 517) data type

Required: No

**Account**

Destination bucket owner account ID. In a cross-account scenario, if you direct Amazon S3 to change replica ownership to the AWS account that owns the destination bucket by specifying the AccessControlTranslation property, this is the account ID of the destination bucket owner. For more information, see Replication Additional Configuration: Changing the Replica Owner in the Amazon Simple Storage Service Developer Guide.

Type: String

Required: No

**Bucket**

The Amazon Resource Name (ARN) of the bucket where you want Amazon S3 to store the results.

Type: String

Required: Yes

**EncryptionConfiguration**

A container that provides information about encryption. If SourceSelectionCriteria is specified, you must specify this element.

Type: EncryptionConfiguration (p. 550) data type

Required: No

**Metrics**

A container specifying replication metrics-related settings enabling replication metrics and events.

Type: Metrics (p. 595) data type

Required: No

**ReplicationTime**

A container specifying S3 Replication Time Control (S3 RTC), including whether S3 RTC is enabled and the time when all objects and operations on objects must be replicated. Must be specified together with a Metrics block.

Type: ReplicationTime (p. 640) data type
Required: No

**StorageClass**

The storage class to use when replicating objects, such as S3 Standard or reduced redundancy. By default, Amazon S3 uses the storage class of the source object to create the object replica.

For valid values, see the `StorageClass` element of the PUT Bucket replication action in the Amazon Simple Storage Service API Reference.

Type: String

Valid Values: STANDARD | REDUCED_REDUNDANCY | STANDARD_IA | ONEZONE_IA | INTELLIGENT_TIERING | GLACIER | DEEP_ARCHIVE | OUTPOSTS

Required: No

**See Also**

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for Ruby V3
Encryption
Service: Amazon Simple Storage Service

Contains the type of server-side encryption used.

Contents

EncryptionType

The server-side encryption algorithm used when storing job results in Amazon S3 (for example, AES256, aws:kms).

Type: String

Valid Values: AES256 | aws:kms

Required: Yes

KMSContext

If the encryption type is aws:kms, this optional value can be used to specify the encryption context for the restore results.

Type: String

Required: No

KMSKeyid

If the encryption type is aws:kms, this optional value specifies the ID of the symmetric customer managed AWS KMS CMK to use for encryption of job results. Amazon S3 only supports symmetric CMKs. For more information, see Using Symmetric and Asymmetric Keys in the AWS Key Management Service Developer Guide.

Type: String

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for Ruby V3
EncryptionConfiguration

Service: Amazon Simple Storage Service

Specifies encryption-related information for an Amazon S3 bucket that is a destination for replicated objects.

Contents

ReplicaKmsKeyID

Specifies the ID (Key ARN or Alias ARN) of the customer managed customer master key (CMK) stored in AWS Key Management Service (KMS) for the destination bucket. Amazon S3 uses this key to encrypt replica objects. Amazon S3 only supports symmetric customer managed CMKs. For more information, see Using Symmetric and Asymmetric Keys in the AWS Key Management Service Developer Guide.

Type: String
Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for Ruby V3
EndEvent
Service: Amazon Simple Storage Service

A message that indicates the request is complete and no more messages will be sent. You should not assume that the request is complete until the client receives an EndEvent.

Contents
The members of this structure are context-dependent.

See Also
For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for Ruby V3
Error
Service: Amazon Simple Storage Service

Container for all error elements.

Contents

Code

The error code is a string that uniquely identifies an error condition. It is meant to be read and understood by programs that detect and handle errors by type.

Amazon S3 error codes

- **Code: AccessDenied**
  - **Description:** Access Denied
  - **HTTP Status Code:** 403 Forbidden
  - **SOAP Fault Code Prefix:** Client
- **Code: AccountProblem**
  - **Description:** There is a problem with your AWS account that prevents the operation from completing successfully. Contact AWS Support for further assistance.
  - **HTTP Status Code:** 403 Forbidden
  - **SOAP Fault Code Prefix:** Client
- **Code: AllAccessDisabled**
  - **Description:** All access to this Amazon S3 resource has been disabled. Contact AWS Support for further assistance.
  - **HTTP Status Code:** 403 Forbidden
  - **SOAP Fault Code Prefix:** Client
- **Code: AmbiguousGrantByEmailAddress**
  - **Description:** The email address you provided is associated with more than one account.
  - **HTTP Status Code:** 400 Bad Request
  - **SOAP Fault Code Prefix:** Client
- **Code: AuthorizationHeaderMalformed**
  - **Description:** The authorization header you provided is invalid.
  - **HTTP Status Code:** 400 Bad Request
  - **HTTP Status Code:** N/A
- **Code: BadDigest**
  - **Description:** The Content-MD5 you specified did not match what we received.
  - **HTTP Status Code:** 400 Bad Request
  - **SOAP Fault Code Prefix:** Client
- **Code: BucketAlreadyExists**
  - **Description:** The requested bucket name is not available. The bucket namespace is shared by all users of the system. Please select a different name and try again.
  - **HTTP Status Code:** 409 Conflict
  - **SOAP Fault Code Prefix:** Client
- **Code: BucketAlreadyOwnedByYou**
  - **Description:** The bucket you tried to create already exists, and you own it. Amazon S3 returns this error in all AWS Regions except in the North Virginia Region. For legacy compatibility, if you re-create an existing bucket that you already own in the North Virginia Region, Amazon S3 returns 200 OK and resets the bucket access control lists (ACLs).
• **Code**: 409 Conflict (in all Regions except the North Virginia Region)
• **SOAP Fault Code Prefix**: Client

• **Code**: bucketNotEmpty
  • **Description**: The bucket you tried to delete is not empty.
  • **HTTP Status Code**: 409 Conflict
  • **SOAP Fault Code Prefix**: Client

• **Code**: CredentialsNotSupported
  • **Description**: This request does not support credentials.
  • **HTTP Status Code**: 400 Bad Request
  • **SOAP Fault Code Prefix**: Client

• **Code**: CrossLocationLoggingProhibited
  • **Description**: Cross-location logging not allowed. Buckets in one geographic location cannot log information to a bucket in another location.
  • **HTTP Status Code**: 403 Forbidden
  • **SOAP Fault Code Prefix**: Client

• **Code**: EntityTooSmall
  • **Description**: Your proposed upload is smaller than the minimum allowed object size.
  • **HTTP Status Code**: 400 Bad Request
  • **SOAP Fault Code Prefix**: Client

• **Code**: EntityTooLarge
  • **Description**: Your proposed upload exceeds the maximum allowed object size.
  • **HTTP Status Code**: 400 Bad Request
  • **SOAP Fault Code Prefix**: Client

• **Code**: ExpiredToken
  • **Description**: The provided token has expired.
  • **HTTP Status Code**: 400 Bad Request
  • **SOAP Fault Code Prefix**: Client

• **Code**: IllegalVersioningConfigurationException
  • **Description**: Indicates that the versioning configuration specified in the request is invalid.
  • **HTTP Status Code**: 400 Bad Request
  • **SOAP Fault Code Prefix**: Client

• **Code**: IncompleteBody
  • **Description**: You did not provide the number of bytes specified by the Content-Length HTTP header
  • **HTTP Status Code**: 400 Bad Request
  • **SOAP Fault Code Prefix**: Client

• **Code**: IncorrectNumberOfFilesInPostRequest
  • **Description**: POST requires exactly one file upload per request.
  • **HTTP Status Code**: 400 Bad Request
  • **SOAP Fault Code Prefix**: Client

• **Code**:InlineDataTooLarge
  • **Description**: Inline data exceeds the maximum allowed size.
  • **HTTP Status Code**: 400 Bad Request
  • **SOAP Fault Code Prefix**: Client

• **Code**: InternalError
• **Description:** We encountered an internal error. Please try again.
  • **HTTP Status Code:** 500 Internal Server Error
  • **SOAP Fault Code Prefix:** Server
  • **Code:** InvalidAccessKeyId
    • **Description:** The AWS access key ID you provided does not exist in our records.
  • **HTTP Status Code:** 403 Forbidden
  • **SOAP Fault Code Prefix:** Client
  • **Code:** InvalidAddressingHeader
    • **Description:** You must specify the Anonymous role.
  • **HTTP Status Code:** N/A
  • **SOAP Fault Code Prefix:** Client
  • **Code:** InvalidArgument
    • **Description:** Invalid Argument
  • **HTTP Status Code:** 400 Bad Request
  • **SOAP Fault Code Prefix:** Client
  • **Code:** InvalidBucketName
    • **Description:** The specified bucket is not valid.
  • **HTTP Status Code:** 400 Bad Request
  • **SOAP Fault Code Prefix:** Client
  • **Code:** InvalidBucketState
    • **Description:** The request is not valid with the current state of the bucket.
  • **HTTP Status Code:** 409 Conflict
  • **SOAP Fault Code Prefix:** Client
  • **Code:** InvalidDigest
    • **Description:** The Content-MD5 you specified is not valid.
  • **HTTP Status Code:** 400 Bad Request
  • **SOAP Fault Code Prefix:** Client
  • **Code:** InvalidEncryptionAlgorithmError
    • **Description:** The encryption request you specified is not valid. The valid value is AES256.
  • **HTTP Status Code:** 400 Bad Request
  • **SOAP Fault Code Prefix:** Client
  • **Code:** InvalidLocationConstraint
    • **Description:** The specified location constraint is not valid. For more information about Regions, see [How to Select a Region for Your Buckets](#).
  • **HTTP Status Code:** 400 Bad Request
  • **SOAP Fault Code Prefix:** Client
  • **Code:** InvalidObjectState
    • **Description:** The operation is not valid for the current state of the object.
  • **HTTP Status Code:** 403 Forbidden
  • **SOAP Fault Code Prefix:** Client
  • **Code:** InvalidPart
    • **Description:** One or more of the specified parts could not be found. The part might not have been uploaded, or the specified entity tag might not have matched the part's entity tag.
  • **HTTP Status Code:** 400 Bad Request
  • **SOAP Fault Code Prefix:** Client
  • **Code:** InvalidPartOrder
• **Description:** The list of parts was not in ascending order. Parts list must be specified in order by part number.
  
  • **HTTP Status Code:** 400 Bad Request
  • **SOAP Fault Code Prefix:** Client
  • **Code:** InvalidPayer
  • **Description:** All access to this object has been disabled. Please contact AWS Support for further assistance.
  • **HTTP Status Code:** 403 Forbidden
  • **SOAP Fault Code Prefix:** Client
  • **Code:** InvalidPolicyDocument
  • **Description:** The content of the form does not meet the conditions specified in the policy document.
  • **HTTP Status Code:** 400 Bad Request
  • **SOAP Fault Code Prefix:** Client
  • **Code:** InvalidRange
  • **Description:** The requested range cannot be satisfied.
  • **HTTP Status Code:** 416 Requested Range Not Satisfiable
  • **SOAP Fault Code Prefix:** Client
  • **Code:** InvalidRequest
  • **Description:** Please use AWS4-HMAC-SHA256.
  • **HTTP Status Code:** 400 Bad Request
  • **Code:** N/A
  • **Code:** InvalidRequest
  • **Description:** SOAP requests must be made over an HTTPS connection.
  • **HTTP Status Code:** 400 Bad Request
  • **SOAP Fault Code Prefix:** Client
  • **Code:** InvalidRequest
  • **Description:** Amazon S3 Transfer Acceleration is not supported for buckets with non-DNS compliant names.
  • **HTTP Status Code:** 400 Bad Request
  • **Code:** N/A
  • **Code:** InvalidRequest
  • **Description:** Amazon S3 Transfer Acceleration is not supported for buckets with periods (.) in their names.
  • **HTTP Status Code:** 400 Bad Request
  • **Code:** N/A
  • **Code:** InvalidRequest
  • **Description:** Amazon S3 Transfer Accelerate endpoint only supports virtual style requests.
  • **HTTP Status Code:** 400 Bad Request
  • **Code:** N/A
  • **Code:** InvalidRequest
  • **Description:** Amazon S3 Transfer Accelerate is not configured on this bucket.
  • **HTTP Status Code:** 400 Bad Request
  • **Code:** N/A
  • **Code:** InvalidRequest
  • **Description:** Amazon S3 Transfer Accelerate is disabled on this bucket.
- **HTTP Status Code:** 400 Bad Request
- **Code:** N/A
- **Code:** InvalidRequest
  - **Description:** Amazon S3 Transfer Acceleration is not supported on this bucket. Contact AWS Support for more information.
  - **HTTP Status Code:** 400 Bad Request
  - **Code:** N/A
- **Code:** InvalidRequest
  - **Description:** Amazon S3 Transfer Acceleration cannot be enabled on this bucket. Contact AWS Support for more information.
  - **HTTP Status Code:** 400 Bad Request
  - **Code:** N/A
- **Code:** InvalidSecurity
  - **Description:** The provided security credentials are not valid.
  - **HTTP Status Code:** 403 Forbidden
  - **SOAP Fault Code Prefix:** Client
- **Code:** InvalidSOAPRequest
  - **Description:** The SOAP request body is invalid.
  - **HTTP Status Code:** 400 Bad Request
  - **SOAP Fault Code Prefix:** Client
- **Code:** InvalidStorageClass
  - **Description:** The storage class you specified is not valid.
  - **HTTP Status Code:** 400 Bad Request
  - **SOAP Fault Code Prefix:** Client
- **Code:** InvalidTargetBucketForLogging
  - **Description:** The target bucket for logging does not exist, is not owned by you, or does not have the appropriate grants for the log-delivery group.
  - **HTTP Status Code:** 400 Bad Request
  - **SOAP Fault Code Prefix:** Client
- **Code:** InvalidToken
  - **Description:** The provided token is malformed or otherwise invalid.
  - **HTTP Status Code:** 400 Bad Request
  - **SOAP Fault Code Prefix:** Client
- **Code:** InvalidURI
  - **Description:** Couldn't parse the specified URI.
  - **HTTP Status Code:** 400 Bad Request
  - **SOAP Fault Code Prefix:** Client
- **Code:** KeyTooLongError
  - **Description:** Your key is too long.
  - **HTTP Status Code:** 400 Bad Request
  - **SOAP Fault Code Prefix:** Client
- **Code:** MalformedACELError
  - **Description:** The XML you provided was not well-formed or did not validate against our published schema.
  - **HTTP Status Code:** 400 Bad Request
  - **SOAP Fault Code Prefix:** Client
• **Code**: MalformedPOSTRequest
  • **Description**: The body of your POST request is not well-formed multipart/form-data.
  • **HTTP Status Code**: 400 Bad Request
  • **SOAP Fault Code Prefix**: Client

• **Code**: MalformedXML
  • **Description**: This happens when the user sends malformed XML (XML that doesn't conform to the published XSD) for the configuration. The error message is, "The XML you provided was not well-formed or did not validate against our published schema."
  • **HTTP Status Code**: 400 Bad Request
  • **SOAP Fault Code Prefix**: Client

• **Code**: MaxMessageLengthExceeded
  • **Description**: Your request was too big.
  • **HTTP Status Code**: 400 Bad Request
  • **SOAP Fault Code Prefix**: Client

• **Code**: MaxPostPreDataLengthExceededError
  • **Description**: Your POST request fields preceding the upload file were too large.
  • **HTTP Status Code**: 400 Bad Request
  • **SOAP Fault Code Prefix**: Client

• **Code**: MetadataTooLarge
  • **Description**: Your metadata headers exceed the maximum allowed metadata size.
  • **HTTP Status Code**: 400 Bad Request
  • **SOAP Fault Code Prefix**: Client

• **Code**: MethodNotAllowed
  • **Description**: The specified method is not allowed against this resource.
  • **HTTP Status Code**: 405 Method Not Allowed
  • **SOAP Fault Code Prefix**: Client

• **Code**: MissingAttachment
  • **Description**: A SOAP attachment was expected, but none were found.
  • **HTTP Status Code**: N/A
  • **SOAP Fault Code Prefix**: Client

• **Code**: MissingRequestBodyError
  • **Description**: This happens when the user sends an empty XML document as a request. The error message is, "Request body is empty."
  • **HTTP Status Code**: 400 Bad Request
  • **SOAP Fault Code Prefix**: Client

• **Code**: MissingSecurityElement
  • **Description**: The SOAP 1.1 request is missing a security element.
  • **HTTP Status Code**: 400 Bad Request
  • **SOAP Fault Code Prefix**: Client

• **Code**: MissingSecurityHeader
  • **Description**: Your request is missing a required header.
  • **HTTP Status Code**: 400 Bad Request
  • **SOAP Fault Code Prefix**: Client
Amazon Simple Storage Service API Reference

Amazon Simple Storage Service

- **SOAP Fault Code Prefix:** Client
  - **Code:** NoSuchLoggingStatusForKey
    - **Description:** There is no such thing as a logging status subresource for a key.
    - **HTTP Status Code:** 400 Bad Request
    - **SOAP Fault Code Prefix:** Client
  - **Code:** NoSuchBucket
    - **Description:** The specified bucket does not exist.
    - **HTTP Status Code:** 404 Not Found
    - **SOAP Fault Code Prefix:** Client
  - **Code:** NoSuchBucketPolicy
    - **Description:** The specified bucket does not have a bucket policy.
    - **HTTP Status Code:** 404 Not Found
    - **SOAP Fault Code Prefix:** Client
  - **Code:** NoSuchKey
    - **Description:** The specified key does not exist.
    - **HTTP Status Code:** 404 Not Found
    - **SOAP Fault Code Prefix:** Client
  - **Code:** NoSuchLifecycleConfiguration
    - **Description:** The lifecycle configuration does not exist.
    - **HTTP Status Code:** 404 Not Found
    - **SOAP Fault Code Prefix:** Client
  - **Code:** NoSuchUpload
    - **Description:** The specified multipart upload does not exist. The upload ID might be invalid, or the multipart upload might have been aborted or completed.
    - **HTTP Status Code:** 404 Not Found
    - **SOAP Fault Code Prefix:** Client
  - **Code:** NoSuchVersion
    - **Description:** Indicates that the version ID specified in the request does not match an existing version.
    - **HTTP Status Code:** 404 Not Found
    - **SOAP Fault Code Prefix:** Client
  - **Code:** NotImplemented
    - **Description:** A header you provided implies functionality that is not implemented.
    - **HTTP Status Code:** 501 Not Implemented
    - **SOAP Fault Code Prefix:** Server
  - **Code:** NotSignedUp
    - **Description:** Your account is not signed up for the Amazon S3 service. You must sign up before you can use Amazon S3. You can sign up at the following URL: https://aws.amazon.com/s3
    - **HTTP Status Code:** 403 Forbidden
    - **SOAP Fault Code Prefix:** Client
  - **Code:** OperationAborted
    - **Description:** A conflicting conditional operation is currently in progress against this resource. Try again.
    - **HTTP Status Code:** 409 Conflict
    - **SOAP Fault Code Prefix:** Client
  - **Code:** PermanentRedirect

API Version 2006-03-01

558
• **Description:** The bucket you are attempting to access must be addressed using the specified endpoint. Send all future requests to this endpoint.
  • **HTTP Status Code:** 301 Moved Permanently
  • **SOAP Fault Code Prefix:** Client
• **Code:** PreconditionFailed
  • **Description:** At least one of the preconditions you specified did not hold.
  • **HTTP Status Code:** 412 Precondition Failed
  • **SOAP Fault Code Prefix:** Client
• **Code:** Redirect
  • **Description:** Temporary redirect.
  • **HTTP Status Code:** 307 Moved Temporarily
  • **SOAP Fault Code Prefix:** Client
• **Code:** RestoreAlreadyInProgress
  • **Description:** Object restore is already in progress.
  • **HTTP Status Code:** 409 Conflict
  • **SOAP Fault Code Prefix:** Client
• **Code:** RequestIsNotMultiPartContent
  • **Description:** Bucket POST must be of the enclosure-type multipart/form-data.
  • **HTTP Status Code:** 400 Bad Request
  • **SOAP Fault Code Prefix:** Client
• **Code:** RequestTimeout
  • **Description:** Your socket connection to the server was not read from or written to within the timeout period.
  • **HTTP Status Code:** 400 Bad Request
  • **SOAP Fault Code Prefix:** Client
• **Code:** RequestTimeTooSkewed
  • **Description:** The difference between the request time and the server's time is too large.
  • **HTTP Status Code:** 403 Forbidden
  • **SOAP Fault Code Prefix:** Client
• **Code:** RequestTorrentOfBucketError
  • **Description:** Requesting the torrent file of a bucket is not permitted.
  • **HTTP Status Code:** 400 Bad Request
  • **SOAP Fault Code Prefix:** Client
• **Code:** SignatureDoesNotMatch
  • **Description:** The request signature we calculated does not match the signature you provided. Check your AWS secret access key and signing method. For more information, see [REST Authentication](#) and [SOAP Authentication](#) for details.
  • **HTTP Status Code:** 403 Forbidden
  • **SOAP Fault Code Prefix:** Client
• **Code:** ServiceUnavailable
  • **Description:** Reduce your request rate.
  • **HTTP Status Code:** 503 Service Unavailable
  • **SOAP Fault Code Prefix:** Server
• **Code:** SlowDown
  • **Description:** Reduce your request rate.
  • **HTTP Status Code:** 503 Slow Down
• **SOAP Fault Code Prefix:** Server
  • **Code:** TemporaryRedirect
    • **Description:** You are being redirected to the bucket while DNS updates.
    • **HTTP Status Code:** 307 Moved Temporarily
  • **SOAP Fault Code Prefix:** Client
    • **Code:** TokenRefreshRequired
      • **Description:** The provided token must be refreshed.
      • **HTTP Status Code:** 400 Bad Request
    • **SOAP Fault Code Prefix:** Client
      • **Code:** TooManyBuckets
        • **Description:** You have attempted to create more buckets than allowed.
        • **HTTP Status Code:** 400 Bad Request
    • **SOAP Fault Code Prefix:** Client
      • **Code:** UnexpectedContent
        • **Description:** This request does not support content.
      • **HTTP Status Code:** 400 Bad Request
      • **SOAP Fault Code Prefix:** Client
      • **Code:** UnresolvableGrantByEmailAddress
        • **Description:** The email address you provided does not match any account on record.
      • **HTTP Status Code:** 400 Bad Request
      • **SOAP Fault Code Prefix:** Client
      • **Code:** UserKeyMustBeSpecified
        • **Description:** The bucket POST must contain the specified field name. If it is specified, check the order of the fields.
      • **HTTP Status Code:** 400 Bad Request
      • **SOAP Fault Code Prefix:** Client

**Key**

The error key.

Type: String

Required: No

**Message**

The error message contains a generic description of the error condition in English. It is intended for a human audience. Simple programs display the message directly to the end user if they encounter an error condition they don't know how or don't care to handle. Sophisticated programs with more exhaustive error handling and proper internationalization are more likely to ignore the error message.

Type: String

Required: No

**VersionId**

The version ID of the error.
Type: String
Required: No

See Also
For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for Ruby V3
ErrorDocument
Service: Amazon Simple Storage Service

The error information.

Contents

Key

The object key name to use when a 4XX class error occurs.

Type: String

Length Constraints: Minimum length of 1.

Required: Yes

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for Ruby V3
ExistingObjectReplication
Service: Amazon Simple Storage Service

Optional configuration to replicate existing source bucket objects. For more information, see Replicating Existing Objects in the Amazon S3 Developer Guide.

Contents

Status

Type: String

Valid Values: Enabled | Disabled

Required: Yes

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for Ruby V3
FilterRule
Service: Amazon Simple Storage Service

Specifies the Amazon S3 object key name to filter on and whether to filter on the suffix or prefix of the key name.

Contents

Name

The object key name prefix or suffix identifying one or more objects to which the filtering rule applies. The maximum length is 1,024 characters. Overlapping prefixes and suffixes are not supported. For more information, see Configuring Event Notifications in the Amazon Simple Storage Service Developer Guide.

Type: String

Valid Values: prefix | suffix

Required: No

Value

The value that the filter searches for in object key names.

Type: String

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for Ruby V3
GlacierJobParameters

Service: Amazon Simple Storage Service

Container for S3 Glacier job parameters.

Contents

Tier

   Retrieval tier at which the restore will be processed.

   Type: String

   Valid Values: Standard | Bulk | Expedited

   Required: Yes

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for Ruby V3
Grant
Service: Amazon Simple Storage Service

Container for grant information.

Contents

Grantee

The person being granted permissions.

Type: Grantee (p. 567) data type

Required: No

Permission

Specifies the permission given to the grantee.

Type: String

Valid Values: FULL_CONTROL | WRITE | WRITE_ACP | READ | READ_ACP

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for Ruby V3
Grantee

Service: Amazon Simple Storage Service

Container for the person being granted permissions.

Contents

DisplayName

Screen name of the grantee.

Type: String

Required: No

EmailAddress

Email address of the grantee.

Note
Using email addresses to specify a grantee is only supported in the following AWS Regions:

- US East (N. Virginia)
- US West (N. California)
- US West (Oregon)
- Asia Pacific (Singapore)
- Asia Pacific (Sydney)
- Asia Pacific (Tokyo)
- Europe (Ireland)
- South America (São Paulo)

For a list of all the Amazon S3 supported Regions and endpoints, see Regions and Endpoints in the AWS General Reference.

Type: String

Required: No

ID

The canonical user ID of the grantee.

Type: String

Required: No

Type

Type of grantee

Type: String

Valid Values: CanonicalUser | AmazonCustomerByEmail | Group

Required: Yes

URI

URI of the grantee group.

Type: String

Required: No
See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for Ruby V3
IndexDocument
Service: Amazon Simple Storage Service

Container for the Suffix element.

Contents

Suffix

A suffix that is appended to a request that is for a directory on the website endpoint (for example, if the suffix is index.html and you make a request to samplebucket/images/ the data that is returned will be for the object with the key name images/index.html) The suffix must not be empty and must not include a slash character.

Type: String

Required: Yes

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for Ruby V3
Initiator
Service: Amazon Simple Storage Service

Container element that identifies who initiated the multipart upload.

Contents

DisplayName

Name of the Principal.

Type: String

Required: No

ID

If the principal is an AWS account, it provides the Canonical User ID. If the principal is an IAM User, it provides a user ARN value.

Type: String

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for Ruby V3
InputSerialization
Service: Amazon Simple Storage Service

Describes the serialization format of the object.

Contents

CompressionType

Specifies object's compression format. Valid values: NONE, GZIP, BZIP2. Default Value: NONE.

Type: String

Valid Values: NONE | GZIP | BZIP2

Required: No

CSV

Describes the serialization of a CSV-encoded object.

Type: CSVInput (p. 538) data type

Required: No

JSON

Specifies JSON as object's input serialization format.

Type: JSONInput (p. 583) data type

Required: No

Parquet

Specifies Parquet as object's input serialization format.

Type: ParquetInput (p. 620) data type

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for Ruby V3
IntelligentTieringAndOperator
Service: Amazon Simple Storage Service

A container for specifying S3 Intelligent-Tiering filters. The filters determine the subset of objects to which the rule applies.

Contents

Prefix

An object key name prefix that identifies the subset of objects to which the configuration applies.

Type: String
Required: No

Tags

All of these tags must exist in the object's tag set in order for the configuration to apply.

Type: Array of Tag (p. 666) data types
Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for Ruby V3
IntelligentTieringConfiguration
Service: Amazon Simple Storage Service

Specifies the S3 Intelligent-Tiering configuration for an Amazon S3 bucket.

For information about the S3 Intelligent-Tiering storage class, see Storage class for automatically optimizing frequently and infrequently accessed objects.

Contents

Filter

Specifies a bucket filter. The configuration only includes objects that meet the filter's criteria.

Type: IntelligentTieringFilter (p. 574) data type

Required: No

Id

The ID used to identify the S3 Intelligent-Tiering configuration.

Type: String

Required: Yes

Status

Specifies the status of the configuration.

Type: String

Valid Values: Enabled | Disabled

Required: Yes

Tierings

Specifies the S3 Intelligent-Tiering storage class tier of the configuration.

Type: Array of Tiering (p. 669) data types

Required: Yes

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for Ruby V3
IntelligentTieringFilter
Service: Amazon Simple Storage Service

The Filter is used to identify objects that the S3 Intelligent-Tiering configuration applies to.

Contents

And

A conjunction (logical AND) of predicates, which is used in evaluating a metrics filter. The operator must have at least two predicates, and an object must match all of the predicates in order for the filter to apply.

Type: IntelligentTieringAndOperator (p. 572) data type

Required: No

Prefix

An object key name prefix that identifies the subset of objects to which the rule applies.

Type: String

Required: No

Tag

A container of a key value name pair.

Type: Tag (p. 666) data type

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for Ruby V3
InventoryConfiguration
Service: Amazon Simple Storage Service

Specifies the inventory configuration for an Amazon S3 bucket. For more information, see GET Bucket inventory in the Amazon Simple Storage Service API Reference.

Contents

Destination

Contains information about where to publish the inventory results.

Type: InventoryDestination (p. 577) data type

Required: Yes

Filter

Specifies an inventory filter. The inventory only includes objects that meet the filter's criteria.

Type: InventoryFilter (p. 579) data type

Required: No

Id

The ID used to identify the inventory configuration.

Type: String

Required: Yes

IncludedObjectVersions

Object versions to include in the inventory list. If set to All, the list includes all the object versions, which adds the version-related fields VersionId, IsLatest, and DeleteMarker to the list. If set to Current, the list does not contain these version-related fields.

Type: String

Valid Values: All | Current

Required: Yes

isEnabled

Specifies whether the inventory is enabled or disabled. If set to True, an inventory list is generated. If set to False, no inventory list is generated.

Type: Boolean

Required: Yes

OptionalFields

Contains the optional fields that are included in the inventory results.

Type: Array of strings

Valid Values: Size | LastModifiedDate | StorageClass | ETag |
IsMultipartUploaded | ReplicationStatus | EncryptionStatus |
ObjectLockRetainUntilDate | ObjectLockMode | ObjectLockLegalHoldStatus | IntelligentTieringAccessTier
Required: No

**Schedule**

Specifies the schedule for generating inventory results.

Type: `InventorySchedule (p. 582)` data type

Required: Yes

**See Also**

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for Ruby V3
InventoryDestination
Service: Amazon Simple Storage Service

规定的存储配置一个Amazon S3 bucket。

Contents

S3BucketDestination

Contains the bucket name, file format, bucket owner (optional), and prefix (optional) where inventory results are published.

Type: InventoryS3BucketDestination (p. 580) data type

Required: Yes

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for Ruby V3
InventoryEncryption
Service: Amazon Simple Storage Service

Contains the type of server-side encryption used to encrypt the inventory results.

Contents

SSEKMS

Specifies the use of SSE-KMS to encrypt delivered inventory reports.
Type: SSEKMS (p. 659) data type
Required: No

SSES3

Specifies the use of SSE-S3 to encrypt delivered inventory reports.
Type: SSES3 (p. 661) data type
Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for Ruby V3
InventoryFilter
Service: Amazon Simple Storage Service

Specifies an inventory filter. The inventory only includes objects that meet the filter's criteria.

Contents

Prefix

The prefix that an object must have to be included in the inventory results.

Type: String
Required: Yes

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for Ruby V3
InventoryS3BucketDestination
Service: Amazon Simple Storage Service

Contains the bucket name, file format, bucket owner (optional), and prefix (optional) where inventory results are published.

Contents

AccountId

The account ID that owns the destination S3 bucket. If no account ID is provided, the owner is not validated before exporting data.

Note

Although this value is optional, we strongly recommend that you set it to help prevent problems if the destination bucket ownership changes.

Type: String

Required: No

Bucket

The Amazon Resource Name (ARN) of the bucket where inventory results will be published.

Type: String

Required: Yes

Encryption

Contains the type of server-side encryption used to encrypt the inventory results.

Type: InventoryEncryption (p. 578) data type

Required: No

Format

Specifies the output format of the inventory results.

Type: String

Valid Values: CSV | ORC | Parquet

Required: Yes

Prefix

The prefix that is prepended to all inventory results.

Type: String

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
• AWS SDK for Ruby V3
InventorySchedule
Service: Amazon Simple Storage Service

Specifies the schedule for generating inventory results.

Contents

Frequency

Specifies how frequently inventory results are produced.

Type: String

Valid Values: Daily | Weekly

Required: Yes

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for Ruby V3
JSONInput
Service: Amazon Simple Storage Service

Specifies JSON as object's input serialization format.

Contents

Type

The type of JSON. Valid values: Document, Lines.

Type: String

Valid Values: DOCUMENT | LINES

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for Ruby V3
JSONOutput
Service: Amazon Simple Storage Service

Specifies JSON as request's output serialization format.

Contents

RecordDelimiter

The value used to separate individual records in the output. If no value is specified, Amazon S3 uses a newline character ('\n').

Type: String
Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for Ruby V3
LambdaFunctionConfiguration
Service: Amazon Simple Storage Service

A container for specifying the configuration for AWS Lambda notifications.

Contents

Events

The Amazon S3 bucket event for which to invoke the AWS Lambda function. For more information, see Supported Event Types in the Amazon Simple Storage Service Developer Guide.

Type: Array of strings


Required: Yes

Filter

Specifies object key name filtering rules. For information about key name filtering, see Configuring Event Notifications in the Amazon Simple Storage Service Developer Guide.

Type: NotificationConfigurationFilter (p. 605) data type

Required: No

Id

An optional unique identifier for configurations in a notification configuration. If you don't provide one, Amazon S3 will assign an ID.

Type: String

Required: No

LambdaFunctionArn

The Amazon Resource Name (ARN) of the AWS Lambda function that Amazon S3 invokes when the specified event type occurs.

Type: String

Required: Yes

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for Ruby V3
LifecycleConfiguration
Service: Amazon Simple Storage Service

Container for lifecycle rules. You can add as many as 1000 rules.

Contents

Rules

Specifies lifecycle configuration rules for an Amazon S3 bucket.

Type: Array of Rule (p. 647) data types

Required: Yes

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for Ruby V3
LifecycleExpiration
Service: Amazon Simple Storage Service

Container for the expiration for the lifecycle of the object.

Contents

Date
Indicates at what date the object is to be moved or deleted. Should be in GMT ISO 8601 Format.

Type: Timestamp
Required: No

Days
Indicates the lifetime, in days, of the objects that are subject to the rule. The value must be a non-zero positive integer.

Type: Integer
Required: No

ExpiredObjectDeleteMarker
Indicates whether Amazon S3 will remove a delete marker with no noncurrent versions. If set to true, the delete marker will be expired; if set to false the policy takes no action. This cannot be specified with Days or Date in a Lifecycle Expiration Policy.

Type: Boolean
Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for Ruby V3
LifecycleRule
Service: Amazon Simple Storage Service

A lifecycle rule for individual objects in an Amazon S3 bucket.

Contents

AbortIncompleteMultipartUpload

Specifies the days since the initiation of an incomplete multipart upload that Amazon S3 will wait before permanently removing all parts of the upload. For more information, see Aborting Incomplete Multipart Uploads Using a Bucket Lifecycle Policy in the Amazon Simple Storage Service Developer Guide.

Type: AbortIncompleteMultipartUpload (p. 514) data type

Required: No

Expiration

Specifies the expiration for the lifecycle of the object in the form of date, days and, whether the object has a delete marker.

Type: LifecycleExpiration (p. 588) data type

Required: No

Filter

The Filter is used to identify objects that a Lifecycle Rule applies to. A Filter must have exactly one of Prefix, Tag, or And specified.

Type: LifecycleRuleFilter (p. 592) data type

Required: No

ID

Unique identifier for the rule. The value cannot be longer than 255 characters.

Type: String

Required: No

NoncurrentVersionExpiration

Specifies when noncurrent object versions expire. Upon expiration, Amazon S3 permanently deletes the noncurrent object versions. You set this lifecycle configuration action on a bucket that has versioning enabled (or suspended) to request that Amazon S3 delete noncurrent object versions at a specific period in the object's lifetime.

Type: NoncurrentVersionExpiration (p. 601) data type

Required: No

NoncurrentVersionTransitions

Specifies the transition rule for the lifecycle rule that describes when noncurrent objects transition to a specific storage class. If your bucket is versioning-enabled (or versioning is suspended), you can set this action to request that Amazon S3 transition noncurrent object versions to a specific storage class at a set period in the object's lifetime.

Type: Array of NoncurrentVersionTransition (p. 602) data types
Prefix

This member has been deprecated.

Prefix identifying one or more objects to which the rule applies. This is No longer used; use Filter instead.

Type: String

Required: No

Status

If 'Enabled', the rule is currently being applied. If 'Disabled', the rule is not currently being applied.

Type: String

Valid Values: Enabled | Disabled

Required: Yes

Transitions

Specifies when an Amazon S3 object transitions to a specified storage class.

Type: Array of Transition (p. 674) data types

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for Ruby V3
LifecycleRuleAndOperator
Service: Amazon Simple Storage Service

This is used in a Lifecycle Rule Filter to apply a logical AND to two or more predicates. The Lifecycle Rule will apply to any object matching all of the predicates configured inside the And operator.

Contents

Prefix

Prefix identifying one or more objects to which the rule applies.

Type: String

Required: No

Tags

All of these tags must exist in the object's tag set in order for the rule to apply.

Type: Array of Tag (p. 666) data types

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for Ruby V3
**LifecycleRuleFilter**  
Service: Amazon Simple Storage Service

The Filter is used to identify objects that a Lifecycle Rule applies to. A Filter must have exactly one of Prefix, Tag, or And specified.

**Contents**

**And**

This is used in a Lifecycle Rule Filter to apply a logical AND to two or more predicates. The Lifecycle Rule will apply to any object matching all of the predicates configured inside the And operator.

Type: LifecycleRuleAndOperator (p. 591) data type

Required: No

**Prefix**

Prefix identifying one or more objects to which the rule applies.

Type: String

Required: No

**Tag**

This tag must exist in the object's tag set in order for the rule to apply.

Type: Tag (p. 666) data type

Required: No

**See Also**

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for Ruby V3
**LoggingEnabled**

Service: Amazon Simple Storage Service

Describes where logs are stored and the prefix that Amazon S3 assigns to all log object keys for a bucket. For more information, see [PUT Bucket logging](https://docs.aws.amazon.com/AmazonS3/latest/API/PUTBucketLogging.html) in the *Amazon Simple Storage Service API Reference*.

**Contents**

**TargetBucket**

Specifies the bucket where you want Amazon S3 to store server access logs. You can have your logs delivered to any bucket that you own, including the same bucket that is being logged. You can also configure multiple buckets to deliver their logs to the same target bucket. In this case, you should choose a different TargetPrefix for each source bucket so that the delivered log files can be distinguished by key.

Type: String

Required: Yes

**TargetGrants**

Container for granting information.

Type: Array of [TargetGrant](https://docs.aws.amazon.com/AmazonS3/latest/API/PutBucketLogging.html) data types

Required: No

**TargetPrefix**

A prefix for all log object keys. If you store log files from multiple Amazon S3 buckets in a single bucket, you can use a prefix to distinguish which log files came from which bucket.

Type: String

Required: Yes

**See Also**

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](https://docs.aws.amazon.com/sdk-for-cpp/v1/developer-guide/s3-putbucketlogging.html)
- [AWS SDK for Go](https://docs.aws.amazon.com/sdk-for-golang/v1/developer-guide/s3-putbucketlogging.html)
- [AWS SDK for Java](https://docs.aws.amazon.com/sdk-for-java-v1/developer-guide/s3-putbucketlogging.html)
- [AWS SDK for Ruby V3](https://docs.aws.amazon.com/sdk-for-ruby/v3/developer-guide/s3-putbucketlogging.html)
**MetadataEntry**

Service: Amazon Simple Storage Service

A metadata key-value pair to store with an object.

**Contents**

**Name**

Name of the Object.

Type: String

Required: No

**Value**

Value of the Object.

Type: String

Required: No

**See Also**

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for Ruby V3
**Metrics**

Service: Amazon Simple Storage Service

A container specifying replication metrics-related settings enabling replication metrics and events.

**Contents**

**EventThreshold**

A container specifying the time threshold for emitting the s3:Replication:OperationMissedThreshold event.

Type: ReplicationTimeValue (p. 641) data type

Required: No

**Status**

Specifies whether the replication metrics are enabled.

Type: String

Valid Values: Enabled | Disabled

Required: Yes

**See Also**

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for Ruby V3
MetricsAndOperator
Service: Amazon Simple Storage Service

A conjunction (logical AND) of predicates, which is used in evaluating a metrics filter. The operator must have at least two predicates, and an object must match all of the predicates in order for the filter to apply.

Contents

Prefix
The prefix used when evaluating an AND predicate.
Type: String
Required: No

Tags
The list of tags used when evaluating an AND predicate.
Type: Array of Tag (p. 666) data types
Required: No

See Also
For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for Ruby V3
MetricsConfiguration
Service: Amazon Simple Storage Service

Specifies a metrics configuration for the CloudWatch request metrics (specified by the metrics configuration ID) from an Amazon S3 bucket. If you're updating an existing metrics configuration, note that this is a full replacement of the existing metrics configuration. If you don't include the elements you want to keep, they are erased. For more information, see PUT Bucket metrics in the Amazon Simple Storage Service API Reference.

Contents

Filter

Specifies a metrics configuration filter. The metrics configuration will only include objects that meet the filter's criteria. A filter must be a prefix, a tag, or a conjunction (MetricsAndOperator).

Type: MetricsFilter (p. 598) data type

Required: No

Id

The ID used to identify the metrics configuration.

Type: String

Required: Yes

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for Ruby V3
MetricsFilter
Service: Amazon Simple Storage Service

Specifies a metrics configuration filter. The metrics configuration only includes objects that meet the filter's criteria. A filter must be a prefix, a tag, or a conjunction (MetricsAndOperator).

Contents

And

A conjunction (logical AND) of predicates, which is used in evaluating a metrics filter. The operator must have at least two predicates, and an object must match all of the predicates in order for the filter to apply.

Type: MetricsAndOperator (p. 596) data type

Required: No

Prefix

The prefix used when evaluating a metrics filter.

Type: String

Required: No

Tag

The tag used when evaluating a metrics filter.

Type: Tag (p. 666) data type

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for Ruby V3
**MultipartUpload**
Service: Amazon Simple Storage Service

Container for the `MultipartUpload` for the Amazon S3 object.

**Contents**

**Initiated**
- Date and time at which the multipart upload was initiated.
- Type: Timestamp
- Required: No

**Initiator**
- Identifies who initiated the multipart upload.
- Type: Initiator (p. 570) data type
- Required: No

**Key**
- Key of the object for which the multipart upload was initiated.
- Type: String
- Length Constraints: Minimum length of 1.
- Required: No

**Owner**
- Specifies the owner of the object that is part of the multipart upload.
- Type: Owner (p. 617) data type
- Required: No

**StorageClass**
- The class of storage used to store the object.
- Type: String
- Valid Values: STANDARD | REDUCED_REDUNDANCY | STANDARD_IA | ONEZONE_IA | INTELLIGENT_TIERING | GLACIER | DEEP_ARCHIVE | OUTPOSTS
- Required: No

**UploadId**
- Upload ID that identifies the multipart upload.
- Type: String
- Required: No

**See Also**

For more information about using this API in one of the language-specific AWS SDKs, see the following:
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for Ruby V3
NoncurrentVersionExpiration
Service: Amazon Simple Storage Service

Specifies when noncurrent object versions expire. Upon expiration, Amazon S3 permanently deletes the noncurrent object versions. You set this lifecycle configuration action on a bucket that has versioning enabled (or suspended) to request that Amazon S3 delete noncurrent object versions at a specific period in the object's lifetime.

Contents

NoncurrentDays

Specifies the number of days an object is noncurrent before Amazon S3 can perform the associated action. For information about the noncurrent days calculations, see How Amazon S3 Calculates When an Object Became Noncurrent in the Amazon Simple Storage Service Developer Guide.

Type: Integer
Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for Ruby V3
NoncurrentVersionTransition
Service: Amazon Simple Storage Service

Container for the transition rule that describes when noncurrent objects transition to the STANDARD_IA, ONEZONE_IA, INTELLIGENT_TIERING, GLACIER, or DEEP_ARCHIVE storage class. If your bucket is versioning-enabled (or versioning is suspended), you can set this action to request that Amazon S3 transition noncurrent object versions to the STANDARD_IA, ONEZONE_IA, INTELLIGENT_TIERING, GLACIER, or DEEP_ARCHIVE storage class at a specific period in the object's lifetime.

Contents

NoncurrentDays

Specifies the number of days an object is noncurrent before Amazon S3 can perform the associated action. For information about the noncurrent days calculations, see How Amazon S3 Calculates How Long an Object Has Been Noncurrent in the Amazon Simple Storage Service Developer Guide.

Type: Integer
Required: No

StorageClass

The class of storage used to store the object.

Type: String

Valid Values: GLACIER | STANDARD_IA | ONEZONE_IA | INTELLIGENT_TIERING | DEEP_ARCHIVE
Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for Ruby V3
NotificationConfiguration
Service: Amazon Simple Storage Service

A container for specifying the notification configuration of the bucket. If this element is empty, notifications are turned off for the bucket.

Contents

LambdaFunctionConfigurations
Describes the AWS Lambda functions to invoke and the events for which to invoke them.

Type: Array of LambdaFunctionConfiguration (p. 585) data types
Required: No

QueueConfigurations
The Amazon Simple Queue Service queues to publish messages to and the events for which to publish messages.

Type: Array of QueueConfiguration (p. 627) data types
Required: No

TopicConfigurations
The topic to which notifications are sent and the events for which notifications are generated.

Type: Array of TopicConfiguration (p. 670) data types
Required: No

See Also
For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for Ruby V3
NotificationsConfigurationDeprecated
Service: Amazon Simple Storage Service

Contents

CloudFunctionConfiguration

Container for specifying the AWS Lambda notification configuration.

Type: CloudFunctionConfiguration (p. 526) data type

Required: No

QueueConfiguration

This data type is deprecated. This data type specifies the configuration for publishing messages to an Amazon Simple Queue Service (Amazon SQS) queue when Amazon S3 detects specified events.

Type: QueueConfigurationDeprecated (p. 629) data type

Required: No

TopicConfiguration

This data type is deprecated. A container for specifying the configuration for publication of messages to an Amazon Simple Notification Service (Amazon SNS) topic when Amazon S3 detects specified events.

Type: TopicConfigurationDeprecated (p. 672) data type

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for Ruby V3
NotificationConfigurationFilter
Service: Amazon Simple Storage Service

Specifies object key name filtering rules. For information about key name filtering, see Configuring Event Notifications in the Amazon Simple Storage Service Developer Guide.

Contents

Key

A container for object key name prefix and suffix filtering rules.

Type: S3KeyFilter (p. 649) data type

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for Ruby V3
Object
Service: Amazon Simple Storage Service

An object consists of data and its descriptive metadata.

Contents

ETag

The entity tag is a hash of the object. The ETag reflects changes only to the contents of an object, not its metadata. The ETag may or may not be an MD5 digest of the object data. Whether or not it is depends on how the object was created and how it is encrypted as described below:

- Objects created by the PUT Object, POST Object, or Copy operation, or through the AWS Management Console, and are encrypted by SSE-S3 or plaintext, have ETags that are an MD5 digest of their object data.
- Objects created by the PUT Object, POST Object, or Copy operation, or through the AWS Management Console, and are encrypted by SSE-C or SSE-KMS, have ETags that are not an MD5 digest of their object data.
- If an object is created by either the Multipart Upload or Part Copy operation, the ETag is not an MD5 digest, regardless of the method of encryption.

Type: String
Required: No

Key

The name that you assign to an object. You use the object key to retrieve the object.

Type: String
Length Constraints: Minimum length of 1.
Required: No

LastModified

The date the Object was Last Modified

Type: Timestamp
Required: No

Owner

The owner of the object

Type: Owner (p. 617) data type
Required: No

Size

Size in bytes of the object

Type: Integer
Required: No

StorageClass

The class of storage used to store the object.
Type: String

Valid Values: STANDARD | REDUCED_REDUNDANCY | GLACIER | STANDARD_IA | ONEZONE_IA | INTELLIGENT_TIERING | DEEP_ARCHIVE | OUTPOSTS

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for Ruby V3
ObjectIdentifier
Service: Amazon Simple Storage Service

Object Identifier is unique value to identify objects.

Contents

Key

Key name of the object to delete.

Type: String

Length Constraints: Minimum length of 1.

Required: Yes

VersionId

VersionId for the specific version of the object to delete.

Type: String

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for Ruby V3
ObjectLockConfiguration
Service: Amazon Simple Storage Service

The container element for Object Lock configuration parameters.

Contents

ObjectLockEnabled
Indicates whether this bucket has an Object Lock configuration enabled.

Type: String

Valid Values: Enabled

Required: No

Rule
The Object Lock rule in place for the specified object.

Type: ObjectLockRule (p. 612) data type

Required: No

See Also
For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for Ruby V3
ObjectLockLegalHold
Service: Amazon Simple Storage Service

A Legal Hold configuration for an object.

Contents

Status

Indicates whether the specified object has a Legal Hold in place.

Type: String

Valid Values: ON | OFF

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for Ruby V3
ObjectLockRetention
Service: Amazon Simple Storage Service

A Retention configuration for an object.

Contents

Mode

Indicates the Retention mode for the specified object.

Type: String

Valid Values: GOVERNANCE | COMPLIANCE

Required: No

RetainUntilDate

The date on which this Object Lock Retention will expire.

Type: Timestamp

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for Ruby V3
**ObjectLockRule**
Service: Amazon Simple Storage Service

The container element for an Object Lock rule.

**Contents**

**DefaultRetention**

The default retention period that you want to apply to new objects placed in the specified bucket.

Type: DefaultRetention (p. 542) data type

Required: No

**See Also**

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for Ruby V3
ObjectVersion
Service: Amazon Simple Storage Service

The version of an object.

Contents

ETag
The entity tag is an MD5 hash of that version of the object.

Type: String
Required: No

IsLatest
Specifies whether the object is (true) or is not (false) the latest version of an object.

Type: Boolean
Required: No

Key
The object key.

Type: String
Length Constraints: Minimum length of 1.
Required: No

LastModified
Date and time the object was last modified.

Type: Timestamp
Required: No

Owner
Specifies the owner of the object.

Type: Owner (p. 617) data type
Required: No

Size
Size in bytes of the object.

Type: Integer
Required: No

StorageClass
The class of storage used to store the object.

Type: String
Valid Values: STANDARD
Required: No
**VersionId**

Version ID of an object.

Type: String

Required: No

**See Also**

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for Ruby V3
OutputLocation
Service: Amazon Simple Storage Service

Describes the location where the restore job's output is stored.

Contents

S3

Describes an S3 location that will receive the results of the restore request.

Type: S3Location (p. 650) data type

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for Ruby V3
OutputSerialization
Service: Amazon Simple Storage Service

Describes how results of the Select job are serialized.

Contents

CSV

Describes the serialization of CSV-encoded Select results.

Type: CSVOutput (p. 540) data type

Required: No

JSON

Specifies JSON as request's output serialization format.

Type: JSONOutput (p. 584) data type

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for Ruby V3
Owner
Service: Amazon Simple Storage Service

Container for the owner's display name and ID.

Contents

DisplayName

Container for the display name of the owner.

Type: String
Required: No

ID

Container for the ID of the owner.

Type: String
Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for Ruby V3
OwnershipControls
Service: Amazon Simple Storage Service

The container element for a bucket's ownership controls.

Contents

Rules

The container element for an ownership control rule.

Type: Array of OwnershipControlsRule (p. 619) data types

Required: Yes

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for Ruby V3
OwnershipControlsRule
Service: Amazon Simple Storage Service

The container element for an ownership control rule.

Contents

ObjectOwnership

The container element for object ownership for a bucket's ownership controls.

BucketOwnerPreferred - Objects uploaded to the bucket change ownership to the bucket owner if the objects are uploaded with the bucket-owner-full-control canned ACL.

ObjectWriter - The uploading account will own the object if the object is uploaded with the bucket-owner-full-control canned ACL.

Type: String

Valid Values: BucketOwnerPreferred | ObjectWriter

Required: Yes

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for Ruby V3
ParquetInput
Service: Amazon Simple Storage Service

Container for Parquet.

Contents

The members of this structure are context-dependent.

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for Ruby V3
Part
Service: Amazon Simple Storage Service

Container for elements related to a part.

Contents

ETag
  Entity tag returned when the part was uploaded.
  Type: String
  Required: No

LastModified
  Date and time at which the part was uploaded.
  Type: Timestamp
  Required: No

PartNumber
  Part number identifying the part. This is a positive integer between 1 and 10,000.
  Type: Integer
  Required: No

Size
  Size in bytes of the uploaded part data.
  Type: Integer
  Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for Ruby V3
PolicyStatus
Service: Amazon Simple Storage Service

The container element for a bucket's policy status.

Contents

IsPublic

The policy status for this bucket. TRUE indicates that this bucket is public. FALSE indicates that the bucket is not public.

Type: Boolean
Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for Ruby V3
Progress
Service: Amazon Simple Storage Service

This data type contains information about progress of an operation.

Contents

BytesProcessed
The current number of uncompressed object bytes processed.
Type: Long
Required: No

BytesReturned
The current number of bytes of records payload data returned.
Type: Long
Required: No

BytesScanned
The current number of object bytes scanned.
Type: Long
Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for Ruby V3
**ProgressEvent**
Service: Amazon Simple Storage Service

This data type contains information about the progress event of an operation.

**Contents**

**Details**

The Progress event details.

Type: Progress (p. 623) data type

Required: No

**See Also**

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for Ruby V3
PublicAccessBlockConfiguration
Service: Amazon Simple Storage Service

The PublicAccessBlock configuration that you want to apply to this Amazon S3 bucket. You can enable the configuration options in any combination. For more information about when Amazon S3 considers a bucket or object public, see The Meaning of “Public” in the Amazon Simple Storage Service Developer Guide.

Contents

BlockPublicAcls

Specifies whether Amazon S3 should block public access control lists (ACLs) for this bucket and objects in this bucket. Setting this element to True causes the following behavior:

- PUT Bucket acl and PUT Object acl calls fail if the specified ACL is public.
- PUT Object calls fail if the request includes a public ACL.
- PUT Bucket calls fail if the request includes a public ACL.

Enabling this setting doesn't affect existing policies or ACLs.

Type: Boolean

Required: No

BlockPublicPolicy

Specifies whether Amazon S3 should block public bucket policies for this bucket. Setting this element to True causes Amazon S3 to reject calls to PUT Bucket policy if the specified bucket policy allows public access.

Enabling this setting doesn't affect existing bucket policies.

Type: Boolean

Required: No

IgnorePublicAcls

Specifies whether Amazon S3 should ignore public ACLs for this bucket and objects in this bucket. Setting this element to True causes Amazon S3 to ignore all public ACLs on this bucket and objects in this bucket.

Enabling this setting doesn't affect the persistence of any existing ACLs and doesn't prevent new public ACLs from being set.

Type: Boolean

Required: No

RestrictPublicBuckets

Specifies whether Amazon S3 should restrict public bucket policies for this bucket. Setting this element to True restricts access to this bucket to only AWS service principals and authorized users within this account if the bucket has a public policy.

Enabling this setting doesn't affect previously stored bucket policies, except that public and cross-account access within any public bucket policy, including non-public delegation to specific accounts, is blocked.

Type: Boolean
Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for Ruby V3
QueueConfiguration
Service: Amazon Simple Storage Service

Specifies the configuration for publishing messages to an Amazon Simple Queue Service (Amazon SQS) queue when Amazon S3 detects specified events.

Contents

Events

A collection of bucket events for which to send notifications

Type: Array of strings


Required: Yes

Filter

Specifies object key name filtering rules. For information about key name filtering, see Configuring Event Notifications in the Amazon Simple Storage Service Developer Guide.

Type: NotificationConfigurationFilter (p. 605) data type

Required: No

Id

An optional unique identifier for configurations in a notification configuration. If you don't provide one, Amazon S3 will assign an ID.

Type: String

Required: No

QueueArn

The Amazon Resource Name (ARN) of the Amazon SQS queue to which Amazon S3 publishes a message when it detects events of the specified type.

Type: String

Required: Yes

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for Ruby V3
QueueConfigurationDeprecated
Service: Amazon Simple Storage Service

This data type is deprecated. Use QueueConfiguration for the same purposes. This data type specifies the configuration for publishing messages to an Amazon Simple Queue Service (Amazon SQS) queue when Amazon S3 detects specified events.

Contents

Event

This member has been deprecated.

The bucket event for which to send notifications.

Type: String


Required: No

Events

A collection of bucket events for which to send notifications

Type: Array of strings


Required: No

Id

An optional unique identifier for configurations in a notification configuration. If you don't provide one, Amazon S3 will assign an ID.

Type: String

Required: No

Queue

The Amazon Resource Name (ARN) of the Amazon SQS queue to which Amazon S3 publishes a message when it detects events of the specified type.

Type: String

Required: No
See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for Ruby V3
RecordsEvent
Service: Amazon Simple Storage Service
The container for the records event.

Contents

Payload
The byte array of partial, one or more result records.
Type: Base64-encoded binary data object
Required: No

See Also
For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for Ruby V3
Redirect
Service: Amazon Simple Storage Service

Specifies how requests are redirected. In the event of an error, you can specify a different error code to return.

Contents

HostName

The host name to use in the redirect request.

Type: String
Required: No

HttpRedirectCode

The HTTP redirect code to use on the response. Not required if one of the siblings is present.

Type: String
Required: No

Protocol

Protocol to use when redirecting requests. The default is the protocol that is used in the original request.

Type: String
Valid Values: http | https
Required: No

ReplaceKeyPrefixWith

The object key prefix to use in the redirect request. For example, to redirect requests for all pages with prefix docs/ (objects in the docs/ folder) to documents/, you can set a condition block with KeyPrefixEquals set to docs/ and in the Redirect set ReplaceKeyPrefixWith to /documents. Not required if one of the siblings is present. Can be present only if ReplaceKeyWith is not provided.

Type: String
Required: No

ReplaceKeyWith

The specific object key to use in the redirect request. For example, redirect request to error.html. Not required if one of the siblings is present. Can be present only if ReplaceKeyPrefixWith is not provided.

Type: String
Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for Ruby V3
RedirectAllRequestsTo
Service: Amazon Simple Storage Service

Specifies the redirect behavior of all requests to a website endpoint of an Amazon S3 bucket.

Contents

HostName
Name of the host where requests are redirected.

Type: String
Required: Yes

Protocol
Protocol to use when redirecting requests. The default is the protocol that is used in the original request.

Type: String
Valid Values: http | https
Required: No

See Also
For more information about using this API in one of the language-specific AWS SDKs, see the following:

• AWS SDK for C++
• AWS SDK for Go
• AWS SDK for Java
• AWS SDK for Ruby V3
ReplicationConfiguration
Service: Amazon Simple Storage Service

A container for replication rules. You can add up to 1,000 rules. The maximum size of a replication configuration is 2 MB.

Contents

Role

The Amazon Resource Name (ARN) of the AWS Identity and Access Management (IAM) role that Amazon S3 assumes when replicating objects. For more information, see How to Set Up Replication in the Amazon Simple Storage Service Developer Guide.

Type: String

Required: Yes

Rules

A container for one or more replication rules. A replication configuration must have at least one rule and can contain a maximum of 1,000 rules.

Type: Array of ReplicationRule (p. 636) data types

Required: Yes

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for Ruby V3
ReplicationRule
Service: Amazon Simple Storage Service

Specifies which Amazon S3 objects to replicate and where to store the replicas.

Contents

DeleteMarkerReplication

Specifies whether Amazon S3 replicates delete markers. If you specify a Filter in your replication configuration, you must also include a DeleteMarkerReplication element. If your Filter includes a Tag element, the DeleteMarkerReplication Status must be set to Disabled, because Amazon S3 does not support replicating delete markers for tag-based rules. For an example configuration, see Basic Rule Configuration.

For more information about delete marker replication, see Basic Rule Configuration.

Note
If you are using an earlier version of the replication configuration, Amazon S3 handles replication of delete markers differently. For more information, see Backward Compatibility.

Type: DeleteMarkerReplication (p. 546) data type

Required: No

Destination

A container for information about the replication destination and its configurations including enabling the S3 Replication Time Control (S3 RTC).

Type: Destination (p. 547) data type

Required: Yes

ExistingObjectReplication

Type: ExistingObjectReplication (p. 563) data type

Required: No

Filter

A filter that identifies the subset of objects to which the replication rule applies. A Filter must specify exactly one Prefix, Tag, or an And child element.

Type: ReplicationRuleFilter (p. 639) data type

Required: No

ID

A unique identifier for the rule. The maximum value is 255 characters.

Type: String

Required: No

Prefix

This member has been deprecated.

An object key name prefix that identifies the object or objects to which the rule applies. The maximum prefix length is 1,024 characters. To include all objects in a bucket, specify an empty string.
Priority
The priority associated with the rule. If you specify multiple rules in a replication configuration, Amazon S3 prioritizes the rules to prevent conflicts when filtering. If two or more rules identify the same object based on a specified filter, the rule with higher priority takes precedence. For example:
- Same object quality prefix-based filter criteria if prefixes you specified in multiple rules overlap
- Same object quality tag-based filter criteria specified in multiple rules

For more information, see Replication in the Amazon Simple Storage Service Developer Guide.

SourceSelectionCriteria
A container that describes additional filters for identifying the source objects that you want to replicate. You can choose to enable or disable the replication of these objects. Currently, Amazon S3 supports only the filter that you can specify for objects created with server-side encryption using a customer master key (CMK) stored in AWS Key Management Service (SSE-KMS).

Status
Specifies whether the rule is enabled.

See Also
For more information about using this API in one of the language-specific AWS SDKs, see the following:
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for Ruby V3
**ReplicationRuleAndOperator**

Service: Amazon Simple Storage Service

A container for specifying rule filters. The filters determine the subset of objects to which the rule applies. This element is required only if you specify more than one filter.

For example:

- If you specify both a `Prefix` and a `Tag` filter, wrap these filters in an `And` tag.
- If you specify a filter based on multiple tags, wrap the `Tag` elements in an `And` tag

### Contents

**Prefix**

An object key name prefix that identifies the subset of objects to which the rule applies.

Type: String

Required: No

**Tags**

An array of tags containing key and value pairs.

Type: Array of `Tag (p. 666)` data types

Required: No

### See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for Ruby V3
ReplicationRuleFilter
Service: Amazon Simple Storage Service

A filter that identifies the subset of objects to which the replication rule applies. A filter must specify exactly one Prefix, Tag, or an And child element.

Contents

And

A container for specifying rule filters. The filters determine the subset of objects to which the rule applies. This element is required only if you specify more than one filter. For example:

- If you specify both a Prefix and a Tag filter, wrap these filters in an And tag.
- If you specify a filter based on multiple tags, wrap the Tag elements in an And tag.

Type: ReplicationRuleAndOperator (p. 638) data type

Required: No

Prefix

An object key name prefix that identifies the subset of objects to which the rule applies.

Type: String

Required: No

Tag

A container for specifying a tag key and value.

The rule applies only to objects that have the tag in their tag set.

Type: Tag (p. 666) data type

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for Ruby V3
ReplicationTime
Service: Amazon Simple Storage Service

A container specifying S3 Replication Time Control (S3 RTC) related information, including whether S3 RTC is enabled and the time when all objects and operations on objects must be replicated. Must be specified together with a Metrics block.

Contents

Status
Specifies whether the replication time is enabled.

Type: String

Valid Values: Enabled | Disabled

Required: Yes

Time
A container specifying the time by which replication should be complete for all objects and operations on objects.

Type: ReplicationTimeValue (p. 641) data type

Required: Yes

See Also
For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for Ruby V3
ReplicationTimeValue
Service: Amazon Simple Storage Service

A container specifying the time value for S3 Replication Time Control (S3 RTC) and replication metrics EventThreshold.

Contents

Minutes

Contains an integer specifying time in minutes.

Valid values: 15 minutes.

Type: Integer

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for Ruby V3
RequestPaymentConfiguration
Service: Amazon Simple Storage Service

Contents

Payer

Specifies who pays for the download and request fees.

Type: String

Valid Values: Requester | BucketOwner

Required: Yes

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for Ruby V3
RequestProgress
Service: Amazon Simple Storage Service

Container for specifying if periodic QueryProgress messages should be sent.

Contents

Enabled

Specifies whether periodic QueryProgress frames should be sent. Valid values: TRUE, FALSE. Default value: FALSE.

Type: Boolean

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for Ruby V3
**RestoreRequest**

Service: Amazon Simple Storage Service

Container for restore job parameters.

**Contents**

**Days**

Lifetime of the active copy in days. Do not use with restores that specify `OutputLocation`.

The Days element is required for regular restores, and must not be provided for select requests.

Type: Integer

Required: No

**Description**

The optional description for the job.

Type: String

Required: No

**GlacierJobParameters**

S3 Glacier related parameters pertaining to this job. Do not use with restores that specify `OutputLocation`.

Type: `GlacierJobParameters (p. 565)` data type

Required: No

**OutputLocation**

Describes the location where the restore job's output is stored.

Type: `OutputLocation (p. 615)` data type

Required: No

**SelectParameters**

Describes the parameters for Select job types.

Type: `SelectParameters (p. 654)` data type

Required: No

**Tier**

Retrieval tier at which the restore will be processed.

Type: String

Valid Values: Standard | Bulk | Expedited

Required: No

**Type**

Type of restore request.

Type: String
Valid Values: SELECT

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for Ruby V3
RoutingRule
Service: Amazon Simple Storage Service

Specifies the redirect behavior and when a redirect is applied. For more information about routing rules, see Configuring advanced conditional redirects in the Amazon Simple Storage Service Developer Guide.

Contents

Condition

A container for describing a condition that must be met for the specified redirect to apply. For example, 1. If request is for pages in the /docs folder, redirect to the /documents folder. 2. If request results in HTTP error 4xx, redirect request to another host where you might process the error.

Type: Condition (p. 531) data type

Required: No

Redirect

Container for redirect information. You can redirect requests to another host, to another page, or with another protocol. In the event of an error, you can specify a different error code to return.

Type: Redirect (p. 632) data type

Required: Yes

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for Ruby V3
Rule
Service: Amazon Simple Storage Service

Specifies lifecycle rules for an Amazon S3 bucket. For more information, see Put Bucket Lifecycle Configuration in the Amazon Simple Storage Service API Reference. For examples, see Put Bucket Lifecycle Configuration Examples

Contents

AbortIncompleteMultipartUpload

Specifies the days since the initiation of an incomplete multipart upload that Amazon S3 will wait before permanently removing all parts of the upload. For more information, see Aborting Incomplete Multipart Uploads Using a Bucket Lifecycle Policy in the Amazon Simple Storage Service Developer Guide.

Type: AbortIncompleteMultipartUpload (p. 514) data type

Required: No

Expiration

Specifies the expiration for the lifecycle of the object.

Type: LifecycleExpiration (p. 588) data type

Required: No

ID

Unique identifier for the rule. The value can't be longer than 255 characters.

Type: String

Required: No

NoncurrentVersionExpiration

Specifies when noncurrent object versions expire. Upon expiration, Amazon S3 permanently deletes the noncurrent object versions. You set this lifecycle configuration action on a bucket that has versioning enabled (or suspended) to request that Amazon S3 delete noncurrent object versions at a specific period in the object's lifetime.

Type: NoncurrentVersionExpiration (p. 601) data type

Required: No

NoncurrentVersionTransition

Container for the transition rule that describes when noncurrent objects transition to the STANDARD_IA, ONEZONE_IA, INTELLIGENT_TIERING, GLACIER, or DEEP_ARCHIVE storage class. If your bucket is versioning-enabled (or versioning is suspended), you can set this action to request that Amazon S3 transition noncurrent object versions to the STANDARD_IA, ONEZONE_IA, INTELLIGENT_TIERING, GLACIER, or DEEP_ARCHIVE storage class at a specific period in the object's lifetime.

Type: NoncurrentVersionTransition (p. 602) data type

Required: No

Prefix

Object key prefix that identifies one or more objects to which this rule applies.
Type: String
Required: Yes

**Status**

If *Enabled*, the rule is currently being applied. If *Disabled*, the rule is not currently being applied.

Type: String

Valid Values: *Enabled* | *Disabled*

Required: Yes

**Transition**

Specifies when an object transitions to a specified storage class. For more information about Amazon S3 lifecycle configuration rules, see Transitioning Objects Using Amazon S3 Lifecycle in the Amazon Simple Storage Service Developer Guide.

Type: Transition (p. 674) data type

Required: No

**See Also**

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for Ruby V3
S3KeyFilter
Service: Amazon Simple Storage Service

A container for object key name prefix and suffix filtering rules.

Contents

FilterRules

A list of containers for the key-value pair that defines the criteria for the filter rule.

Type: Array of FilterRule (p. 564) data types

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for Ruby V3
**S3Location**

Service: Amazon Simple Storage Service

Describes an Amazon S3 location that will receive the results of the restore request.

**Contents**

**AccessControlList**

A list of grants that control access to the staged results.

Type: Array of [Grant](#) data types

Required: No

**BucketName**

The name of the bucket where the restore results will be placed.

Type: String

Required: Yes

**CannedACL**

The canned ACL to apply to the restore results.

Type: String

Valid Values: private | public-read | public-read-write | authenticated-read | aws-exec-read | bucket-owner-read | bucket-owner-full-control

Required: No

**Encryption**

Contains the type of server-side encryption used.

Type: [Encryption](#) data type

Required: No

**Prefix**

The prefix that is prepended to the restore results for this request.

Type: String

Required: Yes

**StorageClass**

The class of storage used to store the restore results.

Type: String

Valid Values: STANDARD | REDUCED_REDUNDANCY | STANDARD_IA | ONEZONE_IA | INTELLIGENT_TIERING | GLACIER | DEEP_ARCHIVE | OUTPOSTS

Required: No

**Tagging**

The tag-set that is applied to the restore results.
Type: Tagging (p. 667) data type

Required: No

UserMetadata

A list of metadata to store with the restore results in S3.

Type: Array of MetadataEntry (p. 594) data types

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for Ruby V3
ScanRange
Service: Amazon Simple Storage Service

Specifies the byte range of the object to get the records from. A record is processed when its first byte is contained by the range. This parameter is optional, but when specified, it must not be empty. See RFC 2616, Section 14.35.1 about how to specify the start and end of the range.

Contents

End

Specifies the end of the byte range. This parameter is optional. Valid values: non-negative integers. The default value is one less than the size of the object being queried. If only the End parameter is supplied, it is interpreted to mean scan the last N bytes of the file. For example, `<scanrange><end>50</end></scanrange>` means scan the last 50 bytes.

Type: Long
Required: No

Start

Specifies the start of the byte range. This parameter is optional. Valid values: non-negative integers. The default value is 0. If only start is supplied, it means scan from that point to the end of the file. For example, `<scanrange><start>50</start></scanrange>` means scan from byte 50 until the end of the file.

Type: Long
Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for Ruby V3
**SelectObjectContentEventStream**

Service: Amazon Simple Storage Service

The container for selecting objects from a content event stream.

**Contents**

**Cont**

The Continuation Event.

Type: `ContinuationEvent (p. 532)` data type

Required: No

**End**

The End Event.

Type: `EndEvent (p. 551)` data type

Required: No

**Progress**

The Progress Event.

Type: `ProgressEvent (p. 624)` data type

Required: No

**Records**

The Records Event.

Type: `RecordsEvent (p. 631)` data type

Required: No

**Stats**

The Stats Event.

Type: `StatsEvent (p. 663)` data type

Required: No

**See Also**

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for Ruby V3
**SelectParameters**
Service: Amazon Simple Storage Service

Describes the parameters for Select job types.

**Contents**

**Expression**

The expression that is used to query the object.

Type: String

Required: Yes

**ExpressionType**

The type of the provided expression (for example, SQL).

Type: String

Valid Values: SQL

Required: Yes

**InputSerialization**

Describes the serialization format of the object.

Type: InputSerialization (p. 571) data type

Required: Yes

**OutputSerialization**

Describes how the results of the Select job are serialized.

Type: OutputSerialization (p. 616) data type

Required: Yes

**See Also**

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Go](#)
- [AWS SDK for Java](#)
- [AWS SDK for Ruby V3](#)
ServerSideEncryptionByDefault
Service: Amazon Simple Storage Service

Describes the default server-side encryption to apply to new objects in the bucket. If a PUT Object request doesn't specify any server-side encryption, this default encryption will be applied. For more information, see PUT Bucket encryption in the Amazon Simple Storage Service API Reference.

Contents

KMSMasterKeyId

AWS Key Management Service (KMS) customer master key ID to use for the default encryption. This parameter is allowed if and only if SSEAlgorithm is set to aws:kms.

You can specify the key ID or the Amazon Resource Name (ARN) of the CMK. However, if you are using encryption with cross-account operations, you must use a fully qualified CMK ARN. For more information, see Using encryption for cross-account operations.

For example:
- Key ID: 1234abcd-12ab-34cd-56ef-1234567890ab
- Key ARN: arn:aws:kms:us-east-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab

Important
Amazon S3 only supports symmetric CMKs and not asymmetric CMKs. For more information, see Using Symmetric and Asymmetric Keys in the AWS Key Management Service Developer Guide.

Type: String
Required: No

SSEAlgorithm

Server-side encryption algorithm to use for the default encryption.

Type: String

Valid Values: AES256 | aws:kms

Required: Yes

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for Ruby V3
ServerSideEncryptionConfiguration
Service: Amazon Simple Storage Service

Specifies the default server-side-encryption configuration.

Contents

Rules

Container for information about a particular server-side encryption configuration rule.

Type: Array of ServerSideEncryptionRule (p. 657) data types

Required: Yes

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for Ruby V3
ServerSideEncryptionRule
Service: Amazon Simple Storage Service

Specifies the default server-side encryption configuration.

Contents

ApplyServerSideEncryptionByDefault

Specifies the default server-side encryption to apply to new objects in the bucket. If a PUT Object request doesn't specify any server-side encryption, this default encryption will be applied.

Type: ServerSideEncryptionByDefault (p. 655) data type

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for Ruby V3
SourceSelectionCriteria
Service: Amazon Simple Storage Service

A container that describes additional filters for identifying the source objects that you want to replicate. You can choose to enable or disable the replication of these objects. Currently, Amazon S3 supports only the filter that you can specify for objects created with server-side encryption using a customer master key (CMK) stored in AWS Key Management Service (SSE-KMS).

Contents

SseKmsEncryptedObjects

A container for filter information for the selection of Amazon S3 objects encrypted with AWS KMS. If you include SourceSelectionCriteria in the replication configuration, this element is required.

Type: SseKmsEncryptedObjects (p. 660) data type

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for Ruby V3
SSEKMS
Service: Amazon Simple Storage Service

Specifies the use of SSE-KMS to encrypt delivered inventory reports.

Contents

KeyId

Specifies the ID of the AWS Key Management Service (AWS KMS) symmetric customer managed customer master key (CMK) to use for encrypting inventory reports.

Type: String
Required: Yes

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for Ruby V3
SseKmsEncryptedObjects
Service: Amazon Simple Storage Service

A container for filter information for the selection of S3 objects encrypted with AWS KMS.

Contents

Status

Specifies whether Amazon S3 replicates objects created with server-side encryption using a customer master key (CMK) stored in AWS Key Management Service.

Type: String

Valid Values: Enabled | Disabled

Required: Yes

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for Ruby V3
**SSES3**
Service: Amazon Simple Storage Service

Specifies the use of SSE-S3 to encrypt delivered inventory reports.

**Contents**

The members of this structure are context-dependent.

**See Also**

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for Ruby V3
Stats
Service: Amazon Simple Storage Service

Container for the stats details.

Contents

BytesProcessed

The total number of uncompressed object bytes processed.

Type: Long

Required: No

BytesReturned

The total number of bytes of records payload data returned.

Type: Long

Required: No

BytesScanned

The total number of object bytes scanned.

Type: Long

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for Ruby V3
StatsEvent

Service: Amazon Simple Storage Service

Container for the Stats Event.

Contents

Details

The Stats event details.

Type: Stats (p. 662) data type

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for Ruby V3
StorageClassAnalysis
Service: Amazon Simple Storage Service

Specifies data related to access patterns to be collected and made available to analyze the tradeoffs between different storage classes for an Amazon S3 bucket.

Contents

DataExport

Specifies how data related to the storage class analysis for an Amazon S3 bucket should be exported.

Type: StorageClassAnalysisDataExport (p. 665) data type

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for Ruby V3
StorageClassAnalysisDataExport
Service: Amazon Simple Storage Service

Container for data related to the storage class analysis for an Amazon S3 bucket for export.

Contents

Destination

The place to store the data for analysis.

Type: AnalyticsExportDestination (p. 520) data type

Required: Yes

OutputSchemaVersion

The version of the output schema to use when exporting data. Must be V_1.

Type: String

Valid Values: V_1

Required: Yes

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for Ruby V3
Tag
Service: Amazon Simple Storage Service

A container of a key value name pair.

Contents

Key

Name of the object key.

Type: String

Length Constraints: Minimum length of 1.

Required: Yes

Value

Value of the tag.

Type: String

Required: Yes

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for Ruby V3
Tagging
Service: Amazon Simple Storage Service

Container for TagSet elements.

Contents

TagSet

A collection for a set of tags

Type: Array of Tag (p. 666) data types

Required: Yes

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for Ruby V3
TargetGrant
Service: Amazon Simple Storage Service

Container for granting information.

Contents

Grantee
Container for the person being granted permissions.

Type: Grantee (p. 567) data type

Required: No

Permission
Logging permissions assigned to the grantee for the bucket.

Type: String

Valid Values: FULL_CONTROL | READ | WRITE

Required: No

See Also
For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for Ruby V3
Tiering
Service: Amazon Simple Storage Service

The S3 Intelligent-Tiering storage class is designed to optimize storage costs by automatically moving data to the most cost-effective storage access tier, without additional operational overhead.

Contents

AccessTier

S3 Intelligent-Tiering access tier. See Storage class for automatically optimizing frequently and infrequently accessed objects for a list of access tiers in the S3 Intelligent-Tiering storage class.

Type: String

Valid Values: ARCHIVE_ACCESS | DEEP_ARCHIVE_ACCESS

Required: Yes

Days

The number of consecutive days of no access after which an object will be eligible to be transitioned to the corresponding tier. The minimum number of days specified for Archive Access tier must be at least 90 days and Deep Archive Access tier must be at least 180 days. The maximum can be up to 2 years (730 days).

Type: Integer

Required: Yes

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for Ruby V3
**TopicConfiguration**

Service: Amazon Simple Storage Service

A container for specifying the configuration for publication of messages to an Amazon Simple Notification Service (Amazon SNS) topic when Amazon S3 detects specified events.

**Contents**

**Events**

The Amazon S3 bucket event about which to send notifications. For more information, see [Supported Event Types](#) in the *Amazon Simple Storage Service Developer Guide*.

Type: Array of strings

Valid Values:  
s3:ReducedRedundancyLostObject | s3:ObjectCreated:* |  
s3:ObjectCreated:Put | s3:ObjectCreated:Post | s3:ObjectCreated:Copy  
| s3:ObjectCreated:CompleteMultipartUpload | s3:ObjectCreated:* |  
s3:ObjectRemoved:Delete | s3:ObjectRemoved:DeleteMarkerCreated |  
s3:ObjectRestore:* | s3:ObjectRestore:Post | s3:ObjectRestore:Completed  
| s3:Replication:* | s3:Replication:OperationFailedReplication |  
s3:Replication:OperationNotTracked | s3:Replication:OperationMissedThreshold |  
s3:Replication:OperationReplicatedAfterThreshold

Required: Yes

**Filter**

Specifies object key name filtering rules. For information about key name filtering, see [Configuring Event Notifications](#) in the *Amazon Simple Storage Service Developer Guide*.

Type: NotificationConfigurationFilter (p. 605) data type

Required: No

**Id**

An optional unique identifier for configurations in a notification configuration. If you don't provide one, Amazon S3 will assign an ID.

Type: String

Required: No

**TopicArn**

The Amazon Resource Name (ARN) of the Amazon SNS topic to which Amazon S3 publishes a message when it detects events of the specified type.

Type: String

Required: Yes

**See Also**

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for Ruby V3
**TopicConfigurationDeprecated**  
Service: Amazon Simple Storage Service

A container for specifying the configuration for publication of messages to an Amazon Simple Notification Service (Amazon SNS) topic when Amazon S3 detects specified events. This data type is deprecated. Use TopicConfiguration instead.

**Contents**

**Event**

*This member has been deprecated.*  
Bucket event for which to send notifications.

Type: String

Valid Values:  
s3:ReducedRedundancyLostObject | s3:ObjectCreated:* |  
s3:ObjectCreated:Put | s3:ObjectCreated:Post | s3:ObjectCreated:Copy |  
s3:ObjectCreated:CompleteMultipartUpload | s3:ObjectRemoved:* |  
s3:ObjectRemoved:Delete | s3:ObjectRemoved:DeleteMarkerCreated |  
s3:ObjectRestore:* | s3:ObjectRestore:Post | s3:ObjectRestore:Completed |  
s3:Replication:* | s3:Replication:OperationFailedReplication |  
s3:Replication:OperationNotTracked | s3:Replication:OperationMissedThreshold |  
s3:Replication:OperationReplicatedAfterThreshold

Required: No

**Events**

A collection of events related to objects

Type: Array of strings

Valid Values:  
s3:ReducedRedundancyLostObject | s3:ObjectCreated:* |  
s3:ObjectCreated:Put | s3:ObjectCreated:Post | s3:ObjectCreated:Copy |  
s3:ObjectCreated:CompleteMultipartUpload | s3:ObjectRemoved:* |  
s3:ObjectRemoved:Delete | s3:ObjectRemoved:DeleteMarkerCreated |  
s3:ObjectRestore:* | s3:ObjectRestore:Post | s3:ObjectRestore:Completed |  
s3:Replication:* | s3:Replication:OperationFailedReplication |  
s3:Replication:OperationNotTracked | s3:Replication:OperationMissedThreshold |  
s3:Replication:OperationReplicatedAfterThreshold

Required: No

**Id**

An optional unique identifier for configurations in a notification configuration. If you don't provide one, Amazon S3 will assign an ID.

Type: String

Required: No

**Topic**

Amazon SNS topic to which Amazon S3 will publish a message to report the specified events for the bucket.

Type: String

Required: No
See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for Ruby V3
Transition
Service: Amazon Simple Storage Service

Specifies when an object transitions to a specified storage class. For more information about Amazon S3 lifecycle configuration rules, see Transitioning Objects Using Amazon S3 Lifecycle in the Amazon Simple Storage Service Developer Guide.

Contents

Date

Indicates when objects are transitioned to the specified storage class. The date value must be in ISO 8601 format. The time is always midnight UTC.

Type: Timestamp
Required: No

Days

Indicates the number of days after creation when objects are transitioned to the specified storage class. The value must be a positive integer.

Type: Integer
Required: No

StorageClass

The storage class to which you want the object to transition.

Type: String

Valid Values: GLACIER | STANDARD_IA | ONEZONE_IA | INTELLIGENT_TIERING | DEEP_ARCHIVE
Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for Ruby V3
**VersioningConfiguration**
Service: Amazon Simple Storage Service

Describes the versioning state of an Amazon S3 bucket. For more information, see **PUT Bucket versioning** in the *Amazon Simple Storage Service API Reference*.

**Contents**

**MFADelete**

Specifies whether MFA delete is enabled in the bucket versioning configuration. This element is only returned if the bucket has been configured with MFA delete. If the bucket has never been so configured, this element is not returned.

Type: String

Valid Values: Enabled | Disabled

Required: No

**Status**

The versioning state of the bucket.

Type: String

Valid Values: Enabled | Suspended

Required: No

**See Also**

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for Ruby V3
WebsiteConfiguration
Service: Amazon Simple Storage Service

Specifies website configuration parameters for an Amazon S3 bucket.

Contents

ErrorDocument
The name of the error document for the website.
Type: ErrorDocument (p. 562) data type
Required: No

IndexDocument
The name of the index document for the website.
Type: IndexDocument (p. 569) data type
Required: No

RedirectAllRequestsTo
The redirect behavior for every request to this bucket's website endpoint.
Important
If you specify this property, you can't specify any other property.
Type: RedirectAllRequestsTo (p. 634) data type
Required: No

RoutingRules
Rules that define when a redirect is applied and the redirect behavior.
Type: Array of RoutingRule (p. 646) data types
Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for Ruby V3

AWS S3 Control

The following data types are supported by AWS S3 Control:

- AbortIncompleteMultipartUpload (p. 678)
- AccessPoint (p. 679)
- CreateBucketConfiguration (p. 681)
- JobDescriptor (p. 682)
- JobFailure (p. 685)
- JobListDescriptor (p. 686)
- JobManifest (p. 688)
- JobManifestLocation (p. 689)
- JobManifestSpec (p. 690)
- JobOperation (p. 691)
- JobProgressSummary (p. 693)
- JobReport (p. 694)
- LambdaInvokeOperation (p. 696)
- LifecycleConfiguration (p. 697)
- LifecycleExpiration (p. 698)
- LifecycleRule (p. 699)
- LifecycleRuleAndOperator (p. 701)
- LifecycleRuleFilter (p. 702)
- NoncurrentVersionExpiration (p. 703)
- NoncurrentVersionTransition (p. 704)
- PolicyStatus (p. 705)
- PublicAccessBlockConfiguration (p. 706)
- RegionalBucket (p. 708)
- S3AccessControlList (p. 709)
- S3AccessControlPolicy (p. 710)
- S3CopyObjectOperation (p. 711)
- S3Grant (p. 714)
- S3Grantee (p. 715)
- S3InitiateRestoreObjectOperation (p. 716)
- S3ObjectLockLegalHold (p. 717)
- S3ObjectMetadata (p. 718)
- S3ObjectOwner (p. 720)
- S3Retention (p. 721)
- S3SetObjectAclOperation (p. 722)
- S3SetObjectLegalHoldOperation (p. 723)
- S3SetObjectRetentionOperation (p. 724)
- S3SetObjectTaggingOperation (p. 725)
- S3Tag (p. 726)
- Tagging (p. 727)
- Transition (p. 728)
- VpcConfiguration (p. 729)
AbortIncompleteMultipartUpload
Service: AWS S3 Control

The container for abort incomplete multipart upload

Contents

DaysAfterInitiation

Specifies the number of days after which Amazon S3 aborts an incomplete multipart upload to the Outposts bucket.

Type: Integer

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for Ruby V3
AccessPoint
Service: AWS S3 Control
An access point used to access a bucket.

Contents

AccessPointArn
The ARN for the access point.
Type: String
Required: No

Bucket
The name of the bucket associated with this access point.
Type: String
Required: Yes

Name
The name of this access point.
Type: String
Required: Yes

NetworkOrigin
Indicates whether this access point allows access from the public internet. If VpcConfiguration is specified for this access point, then NetworkOrigin is VPC, and the access point doesn't allow access from the public internet. Otherwise, NetworkOrigin is Internet, and the access point allows access from the public internet, subject to the access point and bucket access policies.
Type: String
Valid Values: Internet | VPC
Required: Yes

VpcConfiguration
The virtual private cloud (VPC) configuration for this access point, if one exists.
Type: VpcConfiguration (p. 729) data type
Required: No

See Also
For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for Ruby V3
CreateBucketConfiguration

Service: AWS S3 Control

The container for the bucket configuration.

**Note**
This is not supported by Amazon S3 on Outposts buckets.

Contents

**LocationConstraint**

Specifies the Region where the bucket will be created. If you are creating a bucket on the US East (N. Virginia) Region (us-east-1), you do not need to specify the location.

**Note**
This is not supported by Amazon S3 on Outposts buckets.

Type: String

Valid Values: EU | eu-west-1 | us-west-1 | us-west-2 | ap-south-1 | ap-southeast-1 | ap-southeast-2 | ap-northeast-1 | sa-east-1 | cn-north-1 | eu-central-1

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for Ruby V3
**JobDescriptor**
Service: AWS S3 Control

A container element for the job configuration and status information returned by a `Describe Job` request.

**Contents**

**ConfirmationRequired**
Indicates whether confirmation is required before Amazon S3 begins running the specified job. Confirmation is required only for jobs created through the Amazon S3 console.

Type: Boolean
Required: No

**CreationTime**
A timestamp indicating when this job was created.

Type: Timestamp
Required: No

**Description**
The description for this job, if one was provided in this job's `Create Job` request.

Type: String
Length Constraints: Minimum length of 1. Maximum length of 256.
Required: No

**FailureReasons**
If the specified job failed, this field contains information describing the failure.

Type: Array of `JobFailure` data types
Required: No

**JobArn**
The Amazon Resource Name (ARN) for this job.

Type: String
Pattern: `arn:[^:]+:s3:[a-zA-Z0-9-]+:\d{12}:job/.*`
Required: No

**JobId**
The ID for the specified job.

Type: String
Pattern: `[a-zA-Z0-9\-_\_]`

API Version 2006-03-01
682
Required: No

**Manifest**

The configuration information for the specified job's manifest object.

Type: `JobManifest (p. 688)` data type

Required: No

**Operation**

The operation that the specified job is configured to run on the objects listed in the manifest.

Type: `JobOperation (p. 691)` data type

Required: No

**Priority**

The priority of the specified job.

Type: Integer

Valid Range: Minimum value of 0. Maximum value of 2147483647.

Required: No

**ProgressSummary**

Describes the total number of tasks that the specified job has run, the number of tasks that succeeded, and the number of tasks that failed.

Type: `JobProgressSummary (p. 693)` data type

Required: No

**Report**

Contains the configuration information for the job-completion report if you requested one in the `Create Job` request.

Type: `JobReport (p. 694)` data type

Required: No

**RoleArn**

The Amazon Resource Name (ARN) for the AWS Identity and Access Management (IAM) role assigned to run the tasks for this job.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 2048.

Pattern: `arn:[^:]+:iam::\d{12}:role/.*`

Required: No

**Status**

The current status of the specified job.

Type: String

Valid Values: Active | Cancelled | Cancelling | Complete | Completing | Failed | Failing | New | Paused | Pausing | Preparing | Ready | Suspended
Required: No

**StatusUpdateReason**

The reason for updating the job.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 256.

Required: No

**SuspendedCause**

The reason why the specified job was suspended. A job is only suspended if you create it through the Amazon S3 console. When you create the job, it enters the Suspended state to await confirmation before running. After you confirm the job, it automatically exits the Suspended state.

Type: String


Required: No

**SuspendedDate**

The timestamp when this job was suspended, if it has been suspended.

Type: Timestamp

Required: No

**TerminationDate**

A timestamp indicating when this job terminated. A job's termination date is the date and time when it succeeded, failed, or was canceled.

Type: Timestamp

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for Ruby V3
JobFailure
Service: AWS S3 Control

If this job failed, this element indicates why the job failed.

Contents

FailureCode
The failure code, if any, for the specified job.
Type: String
Length Constraints: Minimum length of 1. Maximum length of 64.
Required: No

FailureReason
The failure reason, if any, for the specified job.
Type: String
Length Constraints: Minimum length of 1. Maximum length of 256.
Required: No

See Also
For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for Ruby V3
JobListDescriptor
Service: AWS S3 Control

Contains the configuration and status information for a single job retrieved as part of a job list.

Contents

CreationTime
A timestamp indicating when the specified job was created.
Type: Timestamp
Required: No

Description
The user-specified description that was included in the specified job's Create Job request.
Type: String
Length Constraints: Minimum length of 1. Maximum length of 256.
Required: No

JobId
The ID for the specified job.
Type: String
Pattern: [a-zA-Z0-9\-\_]+
Required: No

Operation
The operation that the specified job is configured to run on each object listed in the manifest.
Type: String
Valid Values: LambdaInvoke | S3PutObjectCopy | S3PutObjectAcl | S3PutObjectTagging | S3InitiateRestoreObject | S3PutObjectLegalHold | S3PutObjectRetention
Required: No

Priority
The current priority for the specified job.
Type: Integer
Valid Range: Minimum value of 0. Maximum value of 2147483647.
Required: No

ProgressSummary
Describes the total number of tasks that the specified job has run, the number of tasks that succeeded, and the number of tasks that failed.
Type: JobProgressSummary (p. 693) data type
Required: No

**Status**

The specified job’s current status.
Type: String

Valid Values: Active | Cancelled | Cancelling | Complete | Completing | Failed | Failing | New | Paused | Pausing | Preparing | Ready | Suspended

Required: No

**TerminationDate**

A timestamp indicating when the specified job terminated. A job's termination date is the date and time when it succeeded, failed, or was canceled.

Type: Timestamp

Required: No

**See Also**

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for Ruby V3
JobManifest
Service: AWS S3 Control

Contains the configuration information for a job's manifest.

Contents

Location

Contains the information required to locate the specified job's manifest.

Type: JobManifestLocation (p. 689) data type

Required: Yes

Spec

Describes the format of the specified job's manifest. If the manifest is in CSV format, also describes the columns contained within the manifest.

Type: JobManifestSpec (p. 690) data type

Required: Yes

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for Ruby V3
JobManifestLocation
Service: AWS S3 Control

Contains the information required to locate a manifest object.

Contents

ETag

The ETag for the specified manifest object.

Type: String


Required: Yes

ObjectArn

The Amazon Resource Name (ARN) for a manifest object.

Type: String


Pattern: arn:[^:]+:s3:.*

Required: Yes

ObjectVersionId

The optional version ID to identify a specific version of the manifest object.

Type: String


Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for Ruby V3
JobManifestSpec
Service: AWS S3 Control

Describes the format of a manifest. If the manifest is in CSV format, also describes the columns contained within the manifest.

Contents

Fields

If the specified manifest object is in the S3BatchOperations_CSV_20180820 format, this element describes which columns contain the required data.

Type: Array of strings

Valid Values: Ignore | Bucket | Key | VersionId

Required: No

Format

Indicates which of the available formats the specified manifest uses.

Type: String

Valid Values: S3BatchOperations_CSV_20180820 | S3InventoryReport_CSV_20161130

Required: Yes

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for Ruby V3
JobOperation
Service: AWS S3 Control

The operation that you want this job to perform on each object listed in the manifest. For more information about the available operations, see Operations in the Amazon Simple Storage Service Developer Guide.

Contents

LambdaInvoke

Directs the specified job to invoke an AWS Lambda function on each object in the manifest.

Type: LambdaInvokeOperation (p. 696) data type

Required: No

S3InitiateRestoreObject

Directs the specified job to run an Initiate Glacier Restore call on each object in the manifest.

Type: S3InitiateRestoreObjectOperation (p. 716) data type

Required: No

S3PutObjectAcl

Directs the specified job to run a PUT Object acl call on each object in the manifest.

Type: S3SetObjectAclOperation (p. 722) data type

Required: No

S3PutObjectCopy

Directs the specified job to run a PUT Copy object call on each object in the manifest.

Type: S3CopyObjectOperation (p. 711) data type

Required: No

S3PutObjectLegalHold

Contains the configuration for an S3 Object Lock legal hold operation that an S3 Batch Operations job passes each object through to the underlying PutObjectLegalHold API. For more information, see Using S3 Object Lock legal hold with S3 Batch Operations in the Amazon Simple Storage Service Developer Guide.

Type: S3SetObjectLegalHoldOperation (p. 723) data type

Required: No

S3PutObjectRetention

Contains the configuration parameters for the Object Lock retention action for an S3 Batch Operations job. Batch Operations passes each value through to the underlying PutObjectRetention API. For more information, see Using S3 Object Lock retention with S3 Batch Operations in the Amazon Simple Storage Service Developer Guide.

Type: S3SetObjectRetentionOperation (p. 724) data type

Required: No

S3PutObjectTagging

Directs the specified job to run a PUT Object tagging call on each object in the manifest.
Type: `S3SetObjectTaggingOperation (p. 725)` data type

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for Ruby V3
JobProgressSummary  
Service: AWS S3 Control

Describes the total number of tasks that the specified job has started, the number of tasks that succeeded, and the number of tasks that failed.

Contents

NumberOfTasksFailed
Type: Long  
Valid Range: Minimum value of 0.  
Required: No

NumberOfTasksSucceeded
Type: Long  
Valid Range: Minimum value of 0.  
Required: No

TotalNumberOfTasks
Type: Long  
Valid Range: Minimum value of 0.  
Required: No

See Also
For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for Ruby V3
JobReport
Service: AWS S3 Control

Contains the configuration parameters for a job-completion report.

Contents

Bucket

The Amazon Resource Name (ARN) for the bucket where specified job-completion report will be stored.

Type: String


Pattern: arn:[^:]+:s3:*

Required: No

Enabled

Indicates whether the specified job will generate a job-completion report.

Type: Boolean

Required: Yes

Format

The format of the specified job-completion report.

Type: String

Valid Values: Report_CSV_20180820

Required: No

Prefix

An optional prefix to describe where in the specified bucket the job-completion report will be stored. Amazon S3 stores the job-completion report at <prefix>/job<-job-id>/report.json.

Type: String


Required: No

ReportScope

Indicates whether the job-completion report will include details of all tasks or only failed tasks.

Type: String

Valid Values: AllTasks | FailedTasksOnly

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:
• AWS SDK for C++
• AWS SDK for Go
• AWS SDK for Java
• AWS SDK for Ruby V3
LambdaInvokeOperation
Service: AWS S3 Control

Contains the configuration parameters for a Lambda Invoke operation.

Contents

FunctionArn

The Amazon Resource Name (ARN) for the AWS Lambda function that the specified job will invoke for each object in the manifest.

Type: String


Pattern: (arn:(aws[a-zA-Z-]*)?:lambda:)?([a-z]([a-z](gov|iso(b?))?[a-z][a-zA-Z0-9-]+:)?(:[a-zA-Z0-9-]+)?(function:)?([a-zA-Z0-9-]+)(?:($LATEST|[a-zA-Z0-9-]+))?

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for Ruby V3
LifecycleConfiguration
Service: AWS S3 Control

The container for the Outposts bucket lifecycle configuration.

Contents

Rules

A lifecycle rule for individual objects in an Outposts bucket.

Type: Array of LifecycleRule (p. 699) data types

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for Ruby V3
**LifecycleExpiration**
Service: AWS S3 Control

The container of the Outposts bucket lifecycle expiration.

**Contents**

**Date**
Indicates at what date the object is to be deleted. Should be in GMT ISO 8601 format.

Type: Timestamp
Required: No

**Days**
Indicates the lifetime, in days, of the objects that are subject to the rule. The value must be a non-zero positive integer.

Type: Integer
Required: No

**ExpiredObjectDeleteMarker**
Indicates whether Amazon S3 will remove a delete marker with no noncurrent versions. If set to true, the delete marker will be expired. If set to false, the policy takes no action. This cannot be specified with Days or Date in a Lifecycle Expiration Policy.

Type: Boolean
Required: No

**See Also**
For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for Ruby V3
LifecycleRule
Service: AWS S3 Control

The container for the Outposts bucket lifecycle rule.

Contents

AbortInCompleteMultipartUpload

Specifies the days since the initiation of an incomplete multipart upload that Amazon S3 waits before permanently removing all parts of the upload. For more information, see Aborting Incomplete Multipart Uploads Using a Bucket Lifecycle Policy in the Amazon Simple Storage Service Developer Guide.

Type: AbortIncompleteMultipartUpload (p. 678) data type

Required: No

Expiration

Specifies the expiration for the lifecycle of the object in the form of date, days and, whether the object has a delete marker.

Type: LifecycleExpiration (p. 698) data type

Required: No

Filter

The container for the filter of lifecycle rule.

Type: LifecycleRuleFilter (p. 702) data type

Required: No

ID

Unique identifier for the rule. The value cannot be longer than 255 characters.

Type: String

Required: No

NoncurrentVersionExpiration

The noncurrent version expiration of the lifecycle rule.

Note
This is not supported by Amazon S3 on Outposts buckets.

Type: NoncurrentVersionExpiration (p. 703) data type

Required: No

NoncurrentVersionTransitions

Specifies the transition rule for the lifecycle rule that describes when noncurrent objects transition to a specific storage class. If your bucket is versioning-enabled (or versioning is suspended), you can set this action to request that Amazon S3 transition noncurrent object versions to a specific storage class at a set period in the object's lifetime.

Note
This is not supported by Amazon S3 on Outposts buckets.

Type: Array of NoncurrentVersionTransition (p. 704) data types
Required: No

**Status**

If 'Enabled', the rule is currently being applied. If 'Disabled', the rule is not currently being applied.

Type: String

Valid Values: Enabled | Disabled

Required: Yes

**Transitions**

Specifies when an Amazon S3 object transitions to a specified storage class.

**Note**

This is not supported by Amazon S3 on Outposts buckets.

Type: Array of Transition (p. 728) data types

Required: No

**See Also**

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for Ruby V3
LifecycleRuleAndOperator
Service: AWS S3 Control

The container for the Outposts bucket lifecycle rule and operator.

Contents

Prefix
Prefix identifying one or more objects to which the rule applies.

Type: String
Required: No

Tags
All of these tags must exist in the object's tag set in order for the rule to apply.

Type: Array of S3Tag (p. 726) data types
Required: No

See Also
For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for Ruby V3
LifecycleRuleFilter
Service: AWS S3 Control
The container for the filter of the lifecycle rule.

Contents

And
The container for the AND condition for the lifecycle rule.
Type: LifecycleRuleAndOperator (p. 701) data type
Required: No

Prefix
Prefix identifying one or more objects to which the rule applies.
Type: String
Required: No

Tag
Type: S3Tag (p. 726) data type
Required: No

See Also
For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for Ruby V3
NoncurrentVersionExpiration
Service: AWS S3 Control

The container of the noncurrent version expiration.

Contents

NoncurrentDays

Specifies the number of days an object is noncurrent before Amazon S3 can perform the associated action. For information about the noncurrent days calculations, see How Amazon S3 Calculates When an Object Became Noncurrent in the Amazon Simple Storage Service Developer Guide.

Type: Integer
Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for Ruby V3
NoncurrentVersionTransition
Service: AWS S3 Control

The container for the noncurrent version transition.

Contents

NoncurrentDays

Specifies the number of days an object is noncurrent before Amazon S3 can perform the associated action. For information about the noncurrent days calculations, see How Amazon S3 Calculates How Long an Object Has Been Noncurrent in the Amazon Simple Storage Service Developer Guide.

Type: Integer
Required: No

StorageClass

The class of storage used to store the object.

Type: String
Valid Values: GLACIER | STANDARD_IA | ONEZONE_IA | INTELLIGENT_TIERING | DEEP_ARCHIVE

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for Ruby V3
PolicyStatus
Service: AWS S3 Control

Indicates whether this access point policy is public. For more information about how Amazon S3 evaluates policies to determine whether they are public, see The Meaning of "Public" in the Amazon Simple Storage Service Developer Guide.

Contents

IsPublic

Type: Boolean
Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for Ruby V3
PublicAccessBlockConfiguration

Service: AWS S3 Control

The PublicAccessBlock configuration that you want to apply to this Amazon S3 account. You can enable the configuration options in any combination. For more information about when Amazon S3 considers a bucket or object public, see The Meaning of "Public" in the Amazon Simple Storage Service Developer Guide.

This is not supported for Amazon S3 on Outposts.

Contents

BlockPublicAcls

Specifies whether Amazon S3 should block public access control lists (ACLs) for buckets in this account. Setting this element to TRUE causes the following behavior:

- PUT Bucket acl and PUT Object acl calls fail if the specified ACL is public.
- PUT Object calls fail if the request includes a public ACL.
- PUT Bucket calls fail if the request includes a public ACL.

Enabling this setting doesn't affect existing policies or ACLs.

This is not supported for Amazon S3 on Outposts.

Type: Boolean

Required: No

BlockPublicPolicy

Specifies whether Amazon S3 should block public bucket policies for buckets in this account. Setting this element to TRUE causes Amazon S3 to reject calls to PUT Bucket policy if the specified bucket policy allows public access.

Enabling this setting doesn't affect existing bucket policies.

This is not supported for Amazon S3 on Outposts.

Type: Boolean

Required: No

IgnorePublicAcls

Specifies whether Amazon S3 should ignore public ACLs for buckets in this account. Setting this element to TRUE causes Amazon S3 to ignore all public ACLs on buckets in this account and any objects that they contain.

Enabling this setting doesn't affect the persistence of any existing ACLs and doesn't prevent new public ACLs from being set.

This is not supported for Amazon S3 on Outposts.

Type: Boolean

Required: No

RestrictPublicBuckets

Specifies whether Amazon S3 should restrict public bucket policies for buckets in this account. Setting this element to TRUE restricts access to buckets with public policies to only AWS service principals and authorized users within this account.
Enabling this setting doesn’t affect previously stored bucket policies, except that public and cross-account access within any public bucket policy, including non-public delegation to specific accounts, is blocked.

This is not supported for Amazon S3 on Outposts.

Type: Boolean

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for Ruby V3
RegionalBucket
Service: AWS S3 Control

The container for the regional bucket.

Contents

Bucket
Type: String


Required: Yes

BucketArn
The Amazon Resource Name (ARN) for the regional bucket.
Type: String


Required: No

CreationDate
The creation date of the regional bucket
Type: Timestamp

Required: Yes

OutpostId
The AWS Outposts ID of the regional bucket.
Type: String

Length Constraints: Minimum length of 1. Maximum length of 64.

Required: No

PublicAccessBlockEnabled
Type: Boolean

Required: Yes

See Also
For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for Ruby V3
S3AccessControlList
Service: AWS S3 Control

Contents

Grants
Type: Array of S3Grant (p. 714) data types
Required: No

Owner
Type: S3ObjectOwner (p. 720) data type
Required: Yes

See Also
For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for Ruby V3
**S3AccessControlPolicy**
Service: AWS S3 Control

**Contents**

**AccessControlList**
Type: [S3AccessControlList](p. 709) data type
Required: No

**CannedAccessControlList**
Type: String
Valid Values: private | public-read | public-read-write | aws-exec-read | authenticated-read | bucket-owner-read | bucket-owner-full-control
Required: No

**See Also**
For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for Ruby V3
S3CopyObjectOperation
Service: AWS S3 Control

Contains the configuration parameters for a PUT Copy object operation. S3 Batch Operations passes each value through to the underlying PUT Copy object API. For more information about the parameters for this operation, see PUT Object - Copy.

Contents

AccessControlGrants
Type: Array of S3Grant (p. 714) data types
Required: No

CannedAccessControlList
Type: String
Valid Values: private | public-read | public-read-write | aws-exec-read | authenticated-read | bucket-owner-read | bucket-owner-full-control
Required: No

MetadataDirective
Type: String
Valid Values: COPY | REPLACE
Required: No

ModifiedSinceConstraint
Type: Timestamp
Required: No

NewObjectMetadata
Type: S3ObjectMetadata (p. 718) data type
Required: No

NewObjectTagging
Type: Array of S3Tag (p. 726) data types
Required: No

ObjectLockLegalHoldStatus
The legal hold status to be applied to all objects in the Batch Operations job.
Type: String
Valid Values: OFF | ON
Required: No

ObjectLockMode
The retention mode to be applied to all objects in the Batch Operations job.
Type: String
Valid Values: COMPLIANCE | GOVERNANCE
Required: No

ObjectLockRetainUntilDate
The date when the applied object retention configuration expires on all objects in the Batch Operations job.
Type: Timestamp
Required: No

RedirectLocation
Specifies an optional metadata property for website redirects, x-amz-website-redirect-location. Allows web page redirects if the object is accessed through a website endpoint.
Type: String
Length Constraints: Minimum length of 1. Maximum length of 2048.
Required: No

RequesterPays
Type: Boolean
Required: No

SSEAwsKmsKeyId
Type: String
Required: No

StorageClass
Type: String
Valid Values: STANDARD | STANDARD_IA | ONEZONE_IA | GLACIER | INTELLIGENT_TIERING | DEEP_ARCHIVE
Required: No

TargetKeyPrefix
Specifies the folder prefix into which you would like the objects to be copied. For example, to copy objects into a folder named "Folder1" in the destination bucket, set the TargetKeyPrefix to "Folder1/".
Type: String
Required: No

TargetResource
Specifies the destination bucket ARN for the batch copy operation. For example, to copy objects to a bucket named "destinationBucket", set the TargetResource to "arn:aws:s3:::destinationBucket".
Type: String
Pattern: arn:\[^:\]+:s3:.*

Required: No

UnModifiedSinceConstraint

Type: Timestamp

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for Ruby V3
S3Grant
Service: AWS S3 Control

Contents

Grantee

Type: S3Grantee (p. 715) data type

Required: No

Permission

Type: String

Valid Values: FULL_CONTROL | READ | WRITE | READ_ACP | WRITE_ACP

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for Ruby V3
S3Grantee
Service: AWS S3 Control

Contents

DisplayName
   Type: String
   Required: No

Identifier
   Type: String
   Required: No

TypeIdentifier
   Type: String
   Valid Values: id | emailAddress | uri
   Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for Ruby V3
S3InitiateRestoreObjectOperation
Service: AWS S3 Control

Contains the configuration parameters for an Initiate Glacier Restore job. S3 Batch Operations passes each value through to the underlying POST Object restore API. For more information about the parameters for this operation, see RestoreObject.

Contents

ExpirationInDays
  Type: Integer
  Valid Range: Minimum value of 0.
  Required: No

GlacierJobTier
  Type: String
  Valid Values: BULK | STANDARD
  Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for Ruby V3
S3ObjectLockLegalHold
Service: AWS S3 Control

Whether S3 Object Lock legal hold will be applied to objects in an S3 Batch Operations job.

Contents

Status

The Object Lock legal hold status to be applied to all objects in the Batch Operations job.

Type: String

Valid Values: OFF | ON

Required: Yes

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for Ruby V3
S3ObjectMetadata
Service: AWS S3 Control

Contents

CacheControl
Type: String
Required: No

ContentDisposition
Type: String
Required: No

ContentEncoding
Type: String
Required: No

ContentLanguage
Type: String
Required: No

ContentLength
Type: Long
Valid Range: Minimum value of 0.
Required: No

ContentMD5
Type: String
Required: No

ContentType
Type: String
Required: No

HttpExpiresDate
Type: Timestamp
Required: No
RequesterCharged

Type: Boolean

Required: No

SSEAlgorithm

Type: String

Valid Values: AES256 | KMS

Required: No

UserMetadata

Type: String to string map

Map Entries: Maximum number of 8192 items.

Key Length Constraints: Minimum length of 1. Maximum length of 1024.

Value Length Constraints: Maximum length of 1024.

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for Ruby V3
S3ObjectOwner
Service: AWS S3 Control

Contents

DisplayName
Type: String
Required: No

ID
Type: String
Required: No

See Also
For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for Ruby V3
S3Retention
Service: AWS S3 Control

Contains the S3 Object Lock retention mode to be applied to all objects in the S3 Batch Operations job. If you don't provide Mode and RetainUntilDate data types in your operation, you will remove the retention from your objects. For more information, see Using S3 Object Lock retention with S3 Batch Operations in the Amazon Simple Storage Service Developer Guide.

Contents

Mode

The Object Lock retention mode to be applied to all objects in the Batch Operations job.

Type: String

Valid Values: COMPLIANCE | GOVERNANCE

Required: No

RetainUntilDate

The date when the applied Object Lock retention will expire on all objects set by the Batch Operations job.

Type: Timestamp

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for Ruby V3
S3SetObjectAclOperation
Service: AWS S3 Control

Contains the configuration parameters for a Set Object ACL operation. S3 Batch Operations passes each value through to the underlying PUT Object acl API. For more information about the parameters for this operation, see PUT Object acl.

Contents

AccessControlPolicy

Type: S3AccessControlPolicy (p. 710) data type
Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for Ruby V3
S3SetObjectLegalHoldOperation
Service: AWS S3 Control

Contains the configuration for an S3 Object Lock legal hold operation that an S3 Batch Operations job passes each object through to the underlying PutObjectLegalHold API. For more information, see Using S3 Object Lock legal hold with S3 Batch Operations in the Amazon Simple Storage Service Developer Guide.

Contents

LegalHold

Contains the Object Lock legal hold status to be applied to all objects in the Batch Operations job.

Type: S3ObjectLockLegalHold (p. 717) data type

Required: Yes

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for Ruby V3
S3SetObjectRetentionOperation
Service: AWS S3 Control

Contains the configuration parameters for the Object Lock retention action for an S3 Batch Operations job. Batch Operations passes each value through to the underlying PutObjectRetention API. For more information, see Using S3 Object Lock retention with S3 Batch Operations in the Amazon Simple Storage Service Developer Guide.

Contents

BypassGovernanceRetention

Indicates if the action should be applied to objects in the Batch Operations job even if they have Object Lock GOVERNANCE type in place.

Type: Boolean
Required: No

Retention

Contains the Object Lock retention mode to be applied to all objects in the Batch Operations job. For more information, see Using S3 Object Lock retention with S3 Batch Operations in the Amazon Simple Storage Service Developer Guide.

Type: S3Retention (p. 721) data type
Required: Yes

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for Ruby V3
S3SetObjectTaggingOperation
Service: AWS S3 Control

Contains the configuration parameters for a Set Object Tagging operation. S3 Batch Operations passes each value through to the underlying PUT Object tagging API. For more information about the parameters for this operation, see PUT Object tagging.

Contents

TagSet

- Type: Array of S3Tag (p. 726) data types
- Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for Ruby V3
S3Tag
Service: AWS S3 Control

Contents

Key
Type: String
Pattern: ^\[[p{L}\p{Z}\p{N}_.:=+\-@%]*\]$ $
Required: Yes

Value
Type: String
Length Constraints: Maximum length of 1024.
Pattern: ^\[[p{L}\p{Z}\p{N}_.:=+\-@%]*\]$ $
Required: Yes

See Also
For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for Ruby V3
Tagging
Service: AWS S3 Control

Contents

TagSet

A collection for a set of tags.

Type: Array of S3Tag (p. 726) data types

Required: Yes

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for Ruby V3
Transition
Service: AWS S3 Control

Specifies when an object transitions to a specified storage class. For more information about Amazon S3 Lifecycle configuration rules, see Transitioning Objects Using Amazon S3 Lifecycle in the Amazon Simple Storage Service Developer Guide.

Contents

Date

Indicates when objects are transitioned to the specified storage class. The date value must be in ISO 8601 format. The time is always midnight UTC.

Type: Timestamp

Required: No

Days

Indicates the number of days after creation when objects are transitioned to the specified storage class. The value must be a positive integer.

Type: Integer

Required: No

StorageClass

The storage class to which you want the object to transition.

Type: String

Valid Values: GLACIER | STANDARD_IA | ONEZONE_IA | INTELLIGENT_TIERING | DEEP_ARCHIVE

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for Ruby V3
VpcConfiguration
Service: AWS S3 Control

The virtual private cloud (VPC) configuration for an access point.

Contents

VpcId

If this field is specified, this access point will only allow connections from the specified VPC ID.

Type: String


Required: Yes

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for Ruby V3

Amazon S3 on Outposts

The following data types are supported by Amazon S3 on Outposts:

- Endpoint (p. 730)
- NetworkInterface (p. 732)
## Endpoint

Service: Amazon S3 on Outposts

Amazon S3 on Outposts access points simplify managing data access at scale for shared datasets in S3 on Outposts. S3 on Outposts uses endpoints to connect to Outposts buckets so that you can perform actions within your virtual private cloud (VPC). For more information, see Using Amazon S3 on Outposts in the Amazon Simple Storage Service Developer Guide.

### Contents

- **CidrBlock**
  - The VPC CIDR committed by this endpoint.
  - Type: String
  - Required: No

- **CreationTime**
  - The time the endpoint was created.
  - Type: Timestamp
  - Required: No

- **EndpointArn**
  - The Amazon Resource Name (ARN) of the endpoint.
  - Type: String
  - Pattern: `^arn:(aws|aws-cn|aws-us-gov|aws-iso|aws-iso-b):s3-outposts:[a-z \-0-9]*:[0-9]{12}:outpost/(op-[a-f0-9]{17}|ec2)/endpoint/[a-zA-Z0-9]{19}$`
  - Required: No

- **NetworkInterfaces**
  - The network interface of the endpoint.
  - Type: Array of NetworkInterface (p. 732) objects
  - Required: No

- **OutpostsId**
  - The ID of the AWS Outposts.
  - Type: String
  - Pattern: `^(op-[a-f0-9]{17}|\d{12}|ec2)$`
  - Required: No

- **Status**
  - The status of the endpoint.

API Version 2006-03-01

730
Type: String

Valid Values: PENDING | AVAILABLE

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for Ruby V3
NetworkInterface
Service: Amazon S3 on Outposts

The container for the network interface.

Contents

NetworkInterfaceId

The ID for the network interface.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 100.

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for Ruby V3
Authenticating Requests (AWS Signature Version 4)

Every interaction with Amazon S3 is either authenticated or anonymous. This section explains request authentication with the AWS Signature Version 4 algorithm.

**Topics**
- Authentication Methods (p. 734)
- Introduction to Signing Requests (p. 734)
- Authenticating Requests: Using the Authorization Header (AWS Signature Version 4) (p. 735)
- Authenticating Requests: Using Query Parameters (AWS Signature Version 4) (p. 755)
- Examples: Signature Calculations in AWS Signature Version 4 (p. 760)
- Authenticating Requests: Browser-Based Uploads Using POST (AWS Signature Version 4) (p. 762)
- Amazon S3 Signature Version 4 Authentication Specific Policy Keys (p. 765)

Authentication with AWS Signature Version 4 provides some or all of the following, depending on how you choose to sign your request:

- **Verification of the identity of the requester** – Authenticated requests require a signature that you create by using your access keys (access key ID, secret access key). For information about getting access keys, see Understanding and Getting Your Security Credentials in the AWS General Reference. If you are using temporary security credentials, the signature calculations also require a security token. For more information, see Requesting Temporary Security Credentials in the IAM User Guide.

- **In-transit data protection** – In order to prevent tampering with a request while it is in transit, you use some of the request elements to calculate the request signature. Upon receiving the request, Amazon S3 calculates the signature by using the same request elements. If any request component received by Amazon S3 does not match the component that was used to calculate the signature, Amazon S3 will reject the request.

- **Protect against reuse of the signed portions of the request** – The signed portions (using AWS Signatures) of requests are valid within 15 minutes of the timestamp in the request. An unauthorized party who has access to a signed request can modify the unsigned portions of the request without affecting the request's validity in the 15 minute window. Because of this, we recommend that you maximize protection by signing request headers and body, making HTTPS requests to Amazon S3, and by using the `x-amz-content-sha256` condition key (see Amazon S3 Signature Version 4 Authentication Specific Policy Keys (p. 765)) in AWS policies to require users to sign Amazon S3 request bodies.

**Note**
Amazon S3 supports Signature Version 4, a protocol for authenticating inbound API requests to AWS services, in all AWS regions. At this time, AWS Regions created before January 30, 2014 will...
Authentication Methods

You can express authentication information by using one of the following methods:

- **HTTP Authorization header** – Using the HTTP Authorization header is the most common method of authenticating an Amazon S3 request. All of the Amazon S3 REST operations (except for browser-based uploads using POST requests) require this header. For more information about the Authorization header value, and how to calculate signature and related options, see Authenticating Requests: Using the Authorization Header (AWS Signature Version 4) (p. 735).

- **Query string parameters** – You can use a query string to express a request entirely in a URL. In this case, you use query parameters to provide request information, including the authentication information. Because the request signature is part of the URL, this type of URL is often referred to as a presigned URL. You can use presigned URLs to embed clickable links, which can be valid for up to seven days, in HTML. For more information, see Authenticating Requests: Using Query Parameters (AWS Signature Version 4) (p. 755).

Amazon S3 also supports browser-based uploads that use HTTP POST requests. With an HTTP POST request, you can upload content to Amazon S3 directly from the browser. For information about authenticating POST requests, see Browser-Based Uploads Using POST in the Amazon Simple Storage Service Developer Guide.

Introduction to Signing Requests

Authentication information that you send in a request must include a signature. To calculate a signature, you first concatenate select request elements to form a string, referred to as the string to sign. You then use a signing key to calculate the hash-based message authentication code (HMAC) of the string to sign.

In AWS Signature Version 4, you don’t use your secret access key to sign the request. Instead, you first use your secret access key to create a signing key. The signing key is scoped to a specific Region and service, and it never expires.

The following diagram illustrates the general process of computing a signature.

1. StringToSign

   A string based on select request elements

2. Signing Key

   - DateKey = HMAC-SHA256(“AWS4-HMAC-HASH"
   - DateRegionKey = HMAC-SHA256(DateKey, “aws-credentials"
   - DateRegionServiceKey = HMAC-SHA256(DateRegionKey, “s3"
   - SigningKey = HMAC-SHA256(DateRegionServiceKey, “s3")

3. Signature

   Signature = HMAC-SHA256(SigningKey, StringToSign)

The string to sign depends on the request type. For example, when you use the HTTP Authorization header or the query parameters for authentication, you use a varying combination of request elements to create the string to sign. For an HTTP POST request, the POST policy in the request is the string you
sign. For more information about computing string to sign, follow links provided at the end of this section.

For signing key, the diagram shows series of calculations, where result of each step you feed into the next step. The final step is the signing key.

Upon receiving an authenticated request, Amazon S3 servers re-create the signature by using the authentication information that is contained in the request. If the signatures match, Amazon S3 processes your request; otherwise, the request is rejected.

For more information about authenticating requests, see the following topics:

- Authenticating Requests: Using the Authorization Header (AWS Signature Version 4) (p. 735)
- Authenticating Requests: Using Query Parameters (AWS Signature Version 4) (p. 755)
- Authenticating Requests in Browser-Based Uploads Using POST (AWS Signature Version 4) (p. 768)

### Authenticating Requests: Using the Authorization Header (AWS Signature Version 4)

#### Topics

- Overview (p. 735)
- Signature Calculations for the Authorization Header: Transferring Payload in a Single Chunk (AWS Signature Version 4) (p. 737)
- Signature Calculations for the Authorization Header: Transferring Payload in Multiple Chunks (Chunked Upload) (AWS Signature Version 4) (p. 748)

#### Overview

Using the HTTP Authorization header is the most common method of providing authentication information. Except for POST requests (p. 769) and requests that are signed by using query parameters, all Amazon S3 operations use the Authorization request header to provide authentication information.

The following is an example of the Authorization header value. Line breaks are added to this example for readability:

```
Authorization: AWS4-HMAC-SHA256
Credential=AKIAIOSFODNN7EXAMPLE/20130524/us-east-1/s3/aws4_request,
SignedHeaders=host;range;x-amz-date,
Signature=fe5f80f77d5fa3beca038a248ff027d0445342fe2855ddc963176630326f1024
```

The following table describes the various components of the Authorization header value in the preceding example:

<table>
<thead>
<tr>
<th>Component</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>AWS4-HMAC-SHA256</td>
<td>The algorithm that was used to calculate the signature. You must provide this value when you use AWS Signature Version 4 for authentication.</td>
</tr>
<tr>
<td></td>
<td>The string specifies AWS Signature Version 4 (AWS4) and the signing algorithm (HMAC-SHA256).</td>
</tr>
</tbody>
</table>
Component | Description
--- | ---
**Credential** | Your access key ID and the scope information, which includes the date, Region, and service that were used to calculate the signature. This string has the following form:

\[
<\text{your-access-key-id}>/<\text{date}>/<\text{aws-region}>/<\text{aws-service}>/aws4_request
\]

Where:
- `<date>` value is specified using `YYYYMMDD` format.
- `<aws-service>` value is `s3` when sending request to Amazon S3.

**SignedHeaders** | A semicolon-separated list of request headers that you used to compute Signature. The list includes header names only, and the header names must be in lowercase. For example:

host;range;x-amz-date

**Signature** | The 256-bit signature expressed as 64 lowercase hexadecimal characters. For example:

fe5f80f77d5fa3beca038a248ff027d0445342fe2855ddc963176630326f1024

Note that the signature calculations vary depending on the option you choose to transfer the payload.

The signature calculations vary depending on the method you choose to transfer the request payload. S3 supports the following options:

- **Transfer payload in a single chunk** – In this case, you have the following signature calculation options:
  - **Signed payload option** – You can optionally compute the entire payload checksum and include it in signature calculation. This provides added security but you need to read your payload twice or buffer it in memory.

    For example, in order to upload a file, you need to read the file first to compute a payload hash for signature calculation and again for transmission when you create the request. For smaller payloads, this approach might be preferable. However, for large files, reading the file twice can be inefficient, so you might want to upload data in chunks instead.

    We recommend you include payload checksum for added security.

  - **Unsigned payload option** – Do not include payload checksum in signature calculation.

    For step-by-step instructions to calculate signature and construct the Authorization header value, see Signature Calculations for the Authorization Header: Transferring Payload in a Single Chunk (AWS Signature Version 4) (p. 737).

- **Transfer payload in multiple chunks (chunked upload)** – In this case you transfer payload in chunks. You can transfer a payload in chunks regardless of the payload size.

    You can break up your payload into chunks. These can be fixed or variable-size chunks. By uploading data in chunks, you avoid reading the entire payload to calculate the signature. Instead, for the first chunk, you calculate a seed signature that uses only the request headers. The second chunk contains...
the signature for the first chunk, and each subsequent chunk contains the signature for the chunk that precedes it. At the end of the upload, you send a final chunk with 0 bytes of data that contains the signature of the last chunk of the payload. For more information, see Signature Calculations for the Authorization Header: Transferring Payload in Multiple Chunks (Chunked Upload) (AWS Signature Version 4) (p. 748).

When you send a request, you must tell Amazon S3 which of the preceding options you have chosen in your signature calculation, by adding the `x-amz-content-sha256` header with one of the following values:

- If you choose chunked upload options, set the header value to `STREAMING-AWS4-HMAC-SHA256-PAYLOAD`.
- If you choose to upload payload in a single chunk, set the header value to the payload checksum (signed payload option), or set the value to the literal string `UNSIGNED-PAYLOAD` (unsigned payload option).

Upon receiving the request, Amazon S3 re-creates the string to sign using information in the Authorization header and the date header. It then verifies with authentication service the signatures match. The request date can be specified by using either the HTTP Date or the `x-amz-date` header. If both headers are present, `x-amz-date` takes precedence.

If the signatures match, Amazon S3 processes your request; otherwise, your request will fail.

For more information, see the following topics:

Signature Calculations for the Authorization Header: Transferring Payload in a Single Chunk (AWS Signature Version 4) (p. 737)

Signature Calculations for the Authorization Header: Transferring Payload in Multiple Chunks (Chunked Upload) (AWS Signature Version 4) (p. 748)

**Signature Calculations for the Authorization Header: Transferring Payload in a Single Chunk (AWS Signature Version 4)**

When using the Authorization header to authenticate requests, the header value includes, among other things, a signature. The signature calculations vary depending on the choice you make for transferring the payload (Overview (p. 735)). This section explains signature calculations when you choose to transfer the payload in a single chunk. The example section (see Examples: Signature Calculations (p. 743)) shows signature calculations and resulting Authorization headers that you can use as a test suite to verify your code.

**Important**

When transferring payload in a single chunk, you can optionally choose to include the payload hash in the signature calculations, referred as `signed payload` (if you don't include it, the payload is considered `unsigned`). The signing procedure discussed in the following section applies to both, but note the following differences:

- **Signed payload option** – You include the payload hash when constructing the canonical request (that then becomes part of StringToSign, as explained in the signature calculation section). You also specify the same value as the `x-amz-content-sha256` header value when sending the request to S3.
- **Unsigned payload option** – You include the literal string `UNSIGNED-PAYLOAD` when constructing a canonical request, and set the same value as the `x-amz-content-sha256` header value when sending the request to Amazon S3.
When you send your request to Amazon S3, the `x-amz-content-sha256` header value informs Amazon S3 whether the payload is signed or not. Amazon S3 can then create signature accordingly for verification.

**Calculating a Signature**

To calculate a signature, you first need a string to sign. You then calculate a HMAC-SHA256 hash of the string to sign by using a signing key. The following diagram illustrates the process, including the various components of the string that you create for signing.

When Amazon S3 receives an authenticated request, it computes the signature and then compares it with the signature that you provided in the request. For that reason, you must compute the signature by using the same method that is used by Amazon S3. The process of putting a request in an agreed-upon form for signing is called canonicalization.

The following table describes the functions that are shown in the diagram. You need to implement code for these functions.

<table>
<thead>
<tr>
<th>Function</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Lowercase()</td>
<td>Convert the string to lowercase.</td>
</tr>
<tr>
<td>Hex()</td>
<td>Lowercase base 16 encoding.</td>
</tr>
<tr>
<td>SHA256Hash()</td>
<td>Secure Hash Algorithm (SHA) cryptographic hash function.</td>
</tr>
<tr>
<td>HMAC-SHA256()</td>
<td>Computes HMAC by using the SHA256 algorithm with the signing key provided. This is the final signature.</td>
</tr>
<tr>
<td>Trim()</td>
<td>Remove any leading or trailing whitespace.</td>
</tr>
<tr>
<td>UriEncode()</td>
<td>URI encode every byte. UriEncode() must enforce the following rules:</td>
</tr>
</tbody>
</table>

API Version 2006-03-01
Amazon Simple Storage Service API Reference
Signature Calculation: Transfer Payload in a Single Chunk

Function

Description
• URI encode every byte except the unreserved characters: 'A'-'Z',
'a'-'z', '0'-'9', '-', '.', '_', and '~'.
• The space character is a reserved character and must be encoded
as "%20" (and not as "+").
• Each URI encoded byte is formed by a '%' and the two-digit
hexadecimal value of the byte.
• Letters in the hexadecimal value must be uppercase, for example
"%1A".
• Encode the forward slash character, '/', everywhere except in the
object key name. For example, if the object key name is photos/
Jan/sample.jpg, the forward slash in the key name is not
encoded.

Important

The standard UriEncode functions provided by your
development platform may not work because of
diﬀerences in implementation and related ambiguity
in the underlying RFCs. We recommend that you write
your own custom UriEncode function to ensure that your
encoding will work.
The following is an example UriEncode() function in Java.
public static String UriEncode(CharSequence input, boolean
encodeSlash) {
StringBuilder result = new StringBuilder();
for (int i = 0; i < input.length(); i++) {
char ch = input.charAt(i);
if ((ch >= 'A' && ch <= 'Z') || (ch >= 'a'
&& ch <= 'z') || (ch >= '0' && ch <= '9') || ch == '_' ||
ch == '-' || ch == '~' || ch == '.') {
result.append(ch);
} else if (ch == '/') {
result.append(encodeSlash ? "%2F" : ch);
} else {
result.append(toHexUTF8(ch));
}
}
return result.toString();
}

Task 1: Create a Canonical Request
This section provides an overview of creating a canonical request.
The following is the canonical request format that Amazon S3 uses to calculate a signature. For
signatures to match, you must create a canonical request in this format:
<HTTPMethod>\n
<CanonicalURI>\n
<CanonicalQueryString>\n
<CanonicalHeaders>\n
<SignedHeaders>\n
<HashedPayload>

API Version 2006-03-01
739


Where:

- **HTTPMethod** is one of the HTTP methods, for example GET, PUT, HEAD, and DELETE.
- **CanonicalURI** is the URI-encoded version of the absolute path component of the URI—everything starting with the "/" that follows the domain name and up to the end of the string or to the question mark character (?) if you have query string parameters. The URI in the following example, `/examplebucket/myphoto.jpg`, is the absolute path and you don't encode the "/" in the absolute path:

  `http://s3.amazonaws.com/examplebucket/myphoto.jpg`

**Note**

You do not normalize URI paths for requests to Amazon S3. For example, you may have a bucket with an object named "my-object/example//photo.user". Normalizing the path changes the object name in the request to "my-object/example/photo.user". This is an incorrect path for that object.

- **CanonicalQueryString** specifies the URI-encoded query string parameters. You URI-encode name and values individually. You must also sort the parameters in the canonical query string alphabetically by key name. The sorting occurs after encoding. The query string in the following URI example is `prefix=somePrefix&marker=someMarker&max-keys=20`:


The canonical query string is as follows (line breaks are added to this example for readability):

```
UriEncode("marker") + "=" + UriEncode("someMarker") + "&" +
UriEncode("max-keys") + "=" + UriEncode("20") + "&" +
UriEncode("prefix") + "=" + UriEncode("somePrefix")
```

When a request targets a subresource, the corresponding query parameter value will be an empty string ("""). For example, the following URI identifies the ACL subresource on the examplebucket bucket:

`http://s3.amazonaws.com/examplebucket?acl`

The CanonicalQueryString in this case is as follows:

```
UriEncode("acl") + "=" + ""
```

If the URI does not include a '?', there is no query string in the request, and you set the canonical query string to an empty string ("""). You will still need to include the "\n".

- **CanonicalHeaders** is a list of request headers with their values. Individual header name and value pairs are separated by the newline character ("\n"). Header names must be in lowercase. You must sort the header names alphabetically to construct the string, as shown in the following example:

```
Lowercase(<HeaderName1>) + "=" + Trim(<value>) + "\n"
Lowercase(<HeaderName2>) + "=" + Trim(<value>) + "\n"
...
Lowercase(<HeaderNameN>) + "=" + Trim(<value>) + "\n"
```

The Lowercase() and Trim() functions used in this example are described in the preceding section.

**The CanonicalHeaders list must include the following:**

API Version 2006-03-01

740
• HTTP host header.
• If the Content-Type header is present in the request, you must add it to the CanonicalHeaders list.
• Any x-amz-* headers that you plan to include in your request must also be added. For example, if you are using temporary security credentials, you need to include x-amz-security-token in your request. You must add this header in the list of CanonicalHeaders.

**Note**
The x-amz-content-sha256 header is required for all AWS Signature Version 4 requests. It provides a hash of the request payload. If there is no payload, you must provide the hash of an empty string.

The following is an example CanonicalHeaders string. The header names are in lowercase and sorted.

```
host:s3.amazonaws.com
x-amz-content-sha256:e3b0c44298fc1c149afbf4c8996fb92427ae41e4649b934ca495991b7852b855
x-amz-date:20130708T220855Z
```

**Note**
For the purpose of calculating an authorization signature, only the host and any x-amz-* headers are required; however, in order to prevent data tampering, you should consider including all the headers in the signature calculation.

• SignedHeaders is an alphabetically sorted, semicolon-separated list of lowercase request header names. The request headers in the list are the same headers that you included in the CanonicalHeaders string. For example, for the previous example, the value of SignedHeaders would be as follows:

```
host;x-amz-content-sha256;x-amz-date
```

• HashedPayload is the hexadecimal value of the SHA256 hash of the request payload.

```
Hex(SHA256Hash(<payload>))
```

If there is no payload in the request, you compute a hash of the empty string as follows:

```
Hex(SHA256Hash(""))
```

The hash returns the following value:

```
e3b0c44298fc1c149afbf4c8996fb92427ae41e4649b934ca495991b7852b855
```

For example, when you upload an object by using a PUT request, you provide object data in the body. When you retrieve an object by using a GET request, you compute the empty string hash.

**Task 2: Create a String to Sign**

This section provides an overview of creating a string to sign. For step-by-step instructions, see Task 2: Create a String to Sign in the AWS General Reference.

The string to sign is a concatenation of the following strings:

```
"AWS4-HMAC-SHA256" + "\n" +
```
**Signature Calculation: Transfer Payload in a Single Chunk**

The constant string `AWS4-HMAC-SHA256` specifies the hash algorithm that you are using, HMAC-SHA256. The `timeStamp` is the current UTC time in ISO 8601 format (for example, `20130524T000000Z`).

`Scope` binds the resulting signature to a specific date, an AWS Region, and a service. Thus, your resulting signature will work only in the specific Region and for a specific service. The signature is valid for seven days after the specified date.

For Amazon S3, the service string is `s3`. For a list of `region` strings, see Regions and Endpoints in the AWS General Reference. The Region column in this table provides the list of valid Region strings.

The following scope restricts the resulting signature to the `us-east-1` Region and Amazon S3.

```
20130606/us-east-1/s3/aws4_request
```

Note

`Scope` must use the same date that you use to compute the signing key, as discussed in the following section.

**Task 3: Calculate Signature**

In AWS Signature Version 4, instead of using your AWS access keys to sign a request, you first create a signing key that is scoped to a specific Region and service. For more information about signing keys, see Introduction to Signing Requests (p. 734).

```
DateKey = HMAC-SHA256("AWS4"+"<SecretAccessKey>", "<YYYYMMDD>")
DateRegionKey = HMAC-SHA256(<DateKey>, "<aws-region>")
DateRegionServiceKey = HMAC-SHA256(<DateRegionKey>, "<aws-service>")
SigningKey = HMAC-SHA256(<DateRegionServiceKey>, "aws4_request")
```

Note

Some use cases can process signature keys for up to 7 days. For more information see Share an Object with Others.

For a list of Region strings, see Regions and Endpoints in the AWS General Reference.

Using a signing key enables you to keep your AWS credentials in one safe place. For example, if you have multiple servers that communicate with Amazon S3, you share the signing key with those servers; you don’t have to keep a copy of your secret access key on each server. Signing key is valid for up to seven days. So each time you calculate signing key you will need to share the signing key with your servers. For more information, see Authenticating Requests (AWS Signature Version 4) (p. 733).

The final signature is the HMAC-SHA256 hash of the string to sign, using the signing key as the key.

```
HMAC-SHA256(SigningKey, StringToSign)
```

For step-by-step instructions on creating a signature, see Task 3: Create a Signature in the AWS General Reference.
Examples: Signature Calculations

You can use the examples in this section as a reference to check signature calculations in your code. For additional references, see Signature Version 4 Test Suite of the AWS General Reference. The calculations shown in the examples use the following data:

- Example access keys.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>AWSAccessKeyId</td>
<td>AKIAIOSFODNN7EXAMPLE</td>
</tr>
<tr>
<td>AWSSecretAccessKey</td>
<td>wJalrXUtnFEMI/K7MDENG/bPxRfiCYEXAMPLEKEY</td>
</tr>
</tbody>
</table>

- Request timestamp of 20130524T000000Z (Fri, 24 May 2013 00:00:00 GMT).
- Bucket name examplebucket.
- The bucket is assumed to be in the US East (N. Virginia) Region. The credential Scope and the Signing Key calculations use us-east-1 as the Region specifier. For information about other Regions, see Regions and Endpoints in the AWS General Reference.
- You can use either path-style or virtual hosted–style requests. The following examples show how to sign a virtual hosted–style request, for example:

```
https://examplebucket.s3.amazonaws.com/photos/photo1.jpg
```

For more information, see Virtual Hosting of Buckets in the Amazon Simple Storage Service Developer Guide.

Example: GET Object

The following example gets the first 10 bytes of an object (test.txt) from examplebucket. For more information about the API action, see GetObject (p. 154).

```
GET /test.txt HTTP/1.1
Host: examplebucket.s3.amazonaws.com
x-amz-date:20130524T000000Z
Authorization: SignatureToBeCalculated
Range: bytes=0-9
x-amz-content-sha256:e3b0c44298fc1c149afbf4c8996fb92427ae41e4649b934ca495991b7852b855
x-amz-date: 20130524T000000Z
```

Because this GET request does not provide any body content, the x-amz-content-sha256 value is the hash of the empty request body. The following steps show signature calculations and construction of the Authorization header.

1. **StringToSign**
   a. **CanonicalRequest**

```
GET /test.txt
host:examplebucket.s3.amazonaws.com
range:bytes=0-9
x-amz-content-sha256:e3b0c44298fc1c149afbf4c8996fb92427ae41e4649b934ca495991b7852b855
x-amz-date:20130524T000000Z
```
In the canonical request string, the last line is the hash of the empty request body. The third line is empty because there are no query parameters in the request.

b. **StringToSign**

```
AWS4-HMAC-SHA256
20130524T000000Z
20130524/us-east-1/s3/aws4_request
7344ae5b7e6c3e7e6b0fe0640412a37625d1fbff95c48bb2dc43964946972
```

2. **SigningKey**

```
signing key = HMAC-SHA256(HMAC-SHA256(HMAC-SHA256(HMAC-SHA256("AWS4" + 
"<YourSecretAccessKey>"),"20130524"),"us-east-1"),"s3"),"aws4_request")
```

3. **Signature**

```
f0e8bdb87c964420e857bd35b5d6ed310bd44f0170aba48dd91039c6036bdb41
```

4. **Authorization header**

The resulting Authorization header is as follows:

```
AWS4-HMAC-SHA256 Credential=AKIAIOSFODNN7EXAMPLE/20130524/us-east-1/s3/aws4_request,SignedHeaders=host;range;x-amz-content-sha256;x-amz-date,Signature=f0e8bdb87c964420e857bd35b5d6ed310bd44f0170aba48dd91039c6036bdb41
```

**Example: PUT Object**

This example PUT request creates an object (test$file.text) in examplebucket. The example assumes the following:

- You are requesting REDUCED_REDUNDANCY as the storage class by adding the `x-amz-storage-class` request header. For information about storage classes, see [Storage Classes](https://aws.amazon.com/aws-developer-blog/2012/09/30/storage-classes-overview/) in the Amazon Simple Storage Service Developer Guide.
- The content of the uploaded file is a string, "Welcome to Amazon S3." The value of `x-amz-content-sha256` in the request is based on this string.

For information about the API action, see [PutObject](https://docs.aws.amazon.com/AmazonS3/latest/API/REST-Object-PUT.html) (p. 343).

```
PUT test$file.text HTTP/1.1
Host: examplebucket.s3.amazonaws.com
Date: Fri, 24 May 2013 00:00:00 GMT
Authorization: SignatureToBeCalculated
x-amz-date: 20130524T000000Z
x-amz-storage-class: REDUCED_REDUNDANCY
x-amz-content-sha256: 44ce7dd67c959e0d3524ffac1771dfbba87d2b6b4e99e42034a8b803f8b072

<Payload>
```

The following steps show signature calculations.
1. **StringToSign**
   
   a. **CanonicalRequest**
   
   ```
   PUT
   /test%24file.text
   
   date:Fri, 24 May 2013 00:00:00 GMT
   host:examplebucket.s3.amazonaws.com
   x-amz-content-sha256:44ce7dd67c959e0d3524ffac1771dfba87d2b6b4e99e420344a8b803f8b072
   x-amz-date:20130524T000000Z
   x-amz-storage-class:REDUCED_REDUNDANCY
   date;host;x-amz-content-sha256;x-amz-date;x-amz-storage-class
   44ce7dd67c959e0d3524ffac1771dfba87d2b6b4e99e420344a8b803f8b072
   ```
   
   In the canonical request, the third line is empty because there are no query parameters in the request. The last line is the hash of the body, which should be same as the `x-amz-content-sha256` header value.

   b. **StringToSign**
   
   ```
   AWS4-HMAC-SHA256
   20130524T000000Z
   20130524/us-east-1/s3/aws4_request
   9e0e9d9c76de8fa5b200d8c849cd5ba8c7a3be3951dbb7f6a76b4158342019d
   ```

2. **SigningKey**

   ```
   signing key = HMAC-SHA256(HMAC-SHA256(HMAC-SHA256(HMAC-SHA256("AWS4" + 
   "<YourSecretAccessKey>"","20130524"),"us-east-1"),"s3"),"aws4_request")
   ```

3. **Signature**

   98ad721746da40c64f1a55b78f14c238d841ea1380cd77a1b5971af0ece108bd

4. **Authorization header**

   The resulting Authorization header is as follows:

   ```
   AWS4-HMAC-SHA256 Credential=AKIAIOSFODNN7EXAMPLE/20130524/us-east-1/s3/aws4_request,SignedHeaders=date;host;x-amz-content-sha256;x-amz-date;x-amz-storage-class,Signature=98ad721746da40c64f1a55b78f14c238d841ea1380cd77a1b5971af0ece108bd
   ```

**Example: GET Bucket Lifecycle**

The following GET request retrieves the lifecycle configuration of examplebucket. For information about the API action, see GetBucketLifecycleConfiguration (p. 113).

```
GET ?lifecycle HTTP/1.1
Host: examplebucket.s3.amazonaws.com
Authorization: SignatureToBeCalculated
x-amz-date: 20130524T000000Z
x-amz-content-sha256:e3b0c44298fc1c149afbf4c8996fb92427ae41e4649b934ca495991b7852b855
```

Because the request does not provide any body content, the `x-amz-content-sha256` header value is the hash of the empty request body. The following steps show signature calculations.
1. **StringToSign**
   a. **CanonicalRequest**

   ```
   GET / lifecycle=
   host:examplebucket.s3.amazonaws.com
   x-amz-content-sha256:e3b0c44298fc1c149afbf4c8996fb92427ae41e4649b934ca495991b7852b855
   x-amz-date:20130524T000000Z
   host;x-amz-content-sha256;x-amz-date
   e3b0c44298fc1c149afbf4c8996fb92427ae41e4649b934ca495991b7852b855
   ```

   In the canonical request, the last line is the hash of the empty request body.

   b. **StringToSign**

   ```
   AWS4-HMAC-SHA256
   20130524T000000Z
   20130524/us-east-1/s3/aws4_request
   9766c79316ff2757b517bc739a67f6213b4ab36dd5da2f94eaebf79c77395ca
   ```

2. **SigningKey**

   ```
   signing key = HMAC-SHA256(HMAC-SHA256(HMAC-SHA256(HMAC-SHA256("AWS4" +
   
   "<YourSecretAccessKey>"),"20130524"),"us-east-1"),"s3"),"aws4_request")
   ```

3. **Signature**

   ```
   f6a454ca298b7da1c68078a5d1dbfbbe0d65c699e0f91ac7a200a0136783543
   ```

4. **Authorization header**

   The resulting Authorization header is as follows:

   ```
   AWS4-HMAC-SHA256 Credential=AKIAIOSFODNN7EXAMPLE/20130524/us-east-1/s3/aws4_request,SignedHeaders=host;x-amz-content-sha256;x-amz-date,Signature=f6a454ca298b7da1c68078a5d1dbfbbe0d65c699e0f91ac7a200a0136783543
   ```

**Example: Get Bucket (List Objects)**

The following example retrieves a list of objects from examplebucket bucket. For information about the API action, see **ListObjects (p. 222).**

```
GET ?max-keys=2&prefix=J HTTP/1.1
Host: examplebucket.s3.amazonaws.com
Authorization: SignatureToBeCalculated
x-amz-date: 20130524T000000Z
x-amz-content-sha256:e3b0c44298fc1c149afbf4c8996fb92427ae41e4649b934ca495991b7852b855
```
1. **StringToSign**
   a. **CanonicalRequest**

   ```
   GET
   /
   max-keys=2&prefix=J
   host:examplebucket.s3.amazonaws.com
   x-amz-content-sha256:e3b0c44298fc1c149afbf4c8996fb92427ae41e4649b934ca495991b7852b855
   x-amz-date:20130524T000000Z
   host;x-amz-content-sha256;x-amz-date
   e3b0c44298fc1c149afbf4c8996fb92427ae41e4649b934ca495991b7852b855
   ```

   In the canonical string, the last line is the hash of the empty request body.
   b. **StringToSign**

   ```
   AWS4-HMAC-SHA256
   20130524T000000Z
   20130524/us-east-1/s3/aws4_request
   df57d21db20da04d7fa30298dd4488ba3a2b47ca3a489c74750e0f1e7df1b9b7
   ```

2. **SigningKey**

   ```
   signing key = HMAC-SHA256(HMAC-SHA256(HMAC-SHA256(HMAC-SHA256("AWS4" + "<YourSecretAccessKey>"),"20130524"),"us-east-1"),"s3"),"aws4_request")
   ```

3. **Signature**

   ```
   34b48302e7b5fa45bde8084f4b7868a86f0a534bc59db6670ed5711ef69dc6f7
   ```

4. **Authorization header**

   The resulting Authorization header is as follows:

   ```
   AWS4-HMAC-SHA256 Credential=AKIAIOSFODNN7EXAMPLE/20130524/us-east-1/s3/aws4_request,SignedHeaders=host;x-amz-content-sha256;x-amz-date,Signature=34b48302e7b5fa45bde8084f4b7868a86f0a534bc59db6670ed5711ef69dc6f7
   ```
Signature Calculations for the Authorization Header: Transferring Payload in Multiple Chunks (Chunked Upload) (AWS Signature Version 4)

As described in the Overview (p. 735), when authenticating requests using the Authorization header, you have an option of uploading the payload in chunks. You can send data in fixed size or variable size chunks. This section describes the signature calculation process in chunked upload, how you create the chunk body, and how the delayed signing works where you first upload the chunk, and send its signature in the subsequent chunk. The example section (see Example: PUT Object (p. 752)) shows signature calculations and resulting Authorization headers that you can use as a test suite to verify your code.

**Note**

When transferring data in a series of chunks, you must do one of the following:

- Explicitly specify the total content length (object length in bytes plus metadata in each chunk) using the Content-Length HTTP header. To do this, you must pre-compute the total length of the payload, including the metadata that you send in each chunk, before starting your request.
- Specify the Transfer-Encoding HTTP header. If you include the Transfer-Encoding header and specify any value other than identity, you must omit the Content-Length header.

For all requests, you must include the x-amz-decoded-content-length header, specifying the size of the object in bytes.

Each chunk signature calculation includes the signature of the previous chunk. To begin, you create a seed signature using only the headers. You use the seed signature in the signature calculation of the first chunk. For each subsequent chunk, you create a chunk signature that includes the signature of the previous chunk. Thus, the chunk signatures are chained together; that is, the signature of chunk \( n \) is a function \( F(chunk \ n, \text{signature}(chunk \ n-1)) \). The chaining ensures that you send the chunks in the correct order.

To perform a chunked upload, do the following:

1. Decide the payload chunk size. You need this when you write the code.
   
   The chunk size must be at least 8 KB. We recommend a chunk size of a least 64 KB for better performance. This chunk size applies to all chunks except the last one. The last chunk you send can be smaller than 8 KB. If your payload is small and can fit into one chunk, then it can be smaller than the 8 KB.

2. Create the seed signature for inclusion in the first chunk. For more information, see Calculating the Seed Signature (p. 748).

3. Create the first chunk and stream it. For more information, see Defining the Chunk Body (p. 751).

4. For each subsequent chunk, calculate the chunk signature that includes the previous signature in the string you sign, construct the chunk, and send it. For more information, see Defining the Chunk Body (p. 751).

5. Send the final additional chunk, which is the same as the other chunks in the construction, but it has zero data bytes. For more information, see Defining the Chunk Body (p. 751).

**Calculating the Seed Signature**

The following diagram illustrates the process of calculating the seed signature.
The following table describes the functions that are shown in the diagram. You need to implement code for these functions.

<table>
<thead>
<tr>
<th>Function</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Lowercase()</td>
<td>Convert the string to lowercase.</td>
</tr>
<tr>
<td>Hex()</td>
<td>Lowercase base 16 encoding.</td>
</tr>
<tr>
<td>SHA256Hash()</td>
<td>Secure Hash Algorithm (SHA) crypticographic hash function.</td>
</tr>
<tr>
<td>HMAC-SHA256()</td>
<td>Computes HMAC by using the SHA256 algorithm with the signing key provided. This is the final signature.</td>
</tr>
<tr>
<td>Trim()</td>
<td>Remove any leading or trailing whitespace.</td>
</tr>
</tbody>
</table>
| UriEncode()        | URI encode every byte. UriEncode() must enforce the following rules:  
|                    |   • URI encode every byte except the unreserved characters: 'A'-'Z',  
|                    |       'a'-'z', '0'-'9', ',', ';', ':', and '\'.  
|                    |   • The space character is a reserved character and must be encoded as "%20" (and not as "+").  
|                    |   • Each URI encoded byte is formed by a '%' and the two-digit hexadecimal value of the byte.  
|                    |   • Letters in the hexadecimal value must be uppercase, for example "%1A".  |
Amazon Simple Storage Service API Reference
Signature Calculation: Transfer Payload in Multiple Chunks

Function

Description
• Encode the forward slash character, '/', everywhere except in the
object key name. For example, if the object key name is photos/
Jan/sample.jpg, the forward slash in the key name is not
encoded.

Important

The standard UriEncode functions provided by your
development platform may not work because of
diﬀerences in implementation and related ambiguity
in the underlying RFCs. We recommend that you write
your own custom UriEncode function to ensure that your
encoding will work.
The following is an example UriEncode() function in Java.
public static String UriEncode(CharSequence input, boolean
encodeSlash) {
StringBuilder result = new StringBuilder();
for (int i = 0; i < input.length(); i++) {
char ch = input.charAt(i);
if ((ch >= 'A' && ch <= 'Z') || (ch >= 'a'
&& ch <= 'z') || (ch >= '0' && ch <= '9') || ch == '_' ||
ch == '-' || ch == '~' || ch == '.') {
result.append(ch);
} else if (ch == '/') {
result.append(encodeSlash ? "%2F" : ch);
} else {
result.append(toHexUTF8(ch));
}
}
return result.toString();
}

For information about the signing process, see Signature Calculations for the Authorization Header:
Transferring Payload in a Single Chunk (AWS Signature Version 4) (p. 737). The process is the same,
except that the creation of CanonicalRequest diﬀers as follows:
• In addition to the request headers you plan to add, you must include the following headers:

Header

Description

x-amz-contentsha256

This header is required for all AWS Signature Version 4 requests. Set the
value to STREAMING-AWS4-HMAC-SHA256-PAYLOAD to indicate that the
signature covers only headers and that there is no payload.

API Version 2006-03-01
750


Content-Encoding | Set the value to `aws-chunked`.
---|---
Amazon S3 supports multiple content encodings. For example:
```
Content-Encoding: aws-chunked, gzip
```
That is, you can specify your custom content-encoding when using Signature Version 4 streaming API.

**Note**
Amazon S3 stores the resulting object without the `aws-chunked` encoding. Therefore, when you retrieve the object, it is not `aws-chunked` encoded.

**x-amz-decoded-content-length** | Set the value to the length, in bytes, of the data to be chunked, without counting any metadata. For example, if you are uploading a 4 GB file, set the value to 4294967296. This is the raw size of the object to be uploaded (data you want to store in Amazon S3).

**Content-Length** | Set the value to the actual size of the transmitted HTTP body, which includes the length of your data (value set for `x-amz-decoded-content-length`), plus chunk metadata. Each chunk has metadata, such as the signature of the previous chunk. Chunk calculations are discussed in the following section. If you include the `Transfer-Encoding` header and specify any value other than `identity`, you must not include the `Content-Length` header.

You send the first chunk with the seed signature. You must construct the chunk as described in the following section.

**Defining the Chunk Body**

All chunks include some metadata. Each chunk must conform to the following structure:

```
string(IntHexBase(chunk-size)) + ";chunk-signatures" + signature + \n + chunk-data + \n
```

Where:

- `IntHexBase()` is a function that you write to convert an integer chunk-size to hexadecimal. For example, if chunk-size is 65536, hexadecimal string is "10000".
- `chunk-size` is the size, in bytes, of the chunk-data, without metadata. For example, if you are uploading a 65 KB object and using a chunk size of 64 KB, you upload the data in three chunks: the first would be 64 KB, the second 1 KB, and the final chunk with 0 bytes.
- `signature` For each chunk, you calculate the signature using the following string to sign. For the first chunk, you use the seed-signature as the previous signature.
The size of the final chunk data that you send is 0, although the chunk body still contains metadata, including the signature of the previous chunk.

**Example: PUT Object**

You can use the examples in this section as a reference to check signature calculations in your code. Before you review the examples, note the following:

- The signature calculations in these examples use the following example security credentials.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>AWSAccessKeyId</td>
<td>AKIAIOSFODNN7EXAMPLE</td>
</tr>
<tr>
<td>AWSSecretAccessKey</td>
<td>wJalrXUtFEMI/K7MDENG/bPxRfiCYEXAMPLEKEY</td>
</tr>
</tbody>
</table>

- All examples use the request timestamp 20130524T000000Z (Fri, 24 May 2013 00:00:00 GMT).
- All examples use examplebucket as the bucket name.
- The bucket is assumed to be in the US East (N. Virginia) Region, and the credential Scope and the Signing Key calculations use us-east-1 as the Region specifier. For more information, see Regions and Endpoints in the Amazon Web Services General Reference.
- You can use either path style or virtual-hosted style requests. The following examples use virtual-hosted style requests, for example:

    https://examplebucket.s3.amazonaws.com/photos/photo1.jpg

For more information, see Virtual Hosting of Buckets in the Amazon Simple Storage Service Developer Guide.

**Example: PUT Object**

The following example sends a PUT request to upload an object. The signature calculations assume the following:

- You are uploading a 65 KB text file, and the file content is a one-character string made up of the letter 'a'.
- The chunk size is 64 KB. As a result, the payload is uploaded in three chunks, 64 KB, 1 KB, and the final chunk with 0 bytes of chunk data.
- The resulting object has the key name chunkObject.txt.
You are requesting REDUCED_REDUNDANCY as the storage class by adding the x-amz-storage-class request header.

For information about the API action, see PutObject (p. 343). The general request syntax is as follows:

```
PUT /examplebucket/chunkObject.txt HTTP/1.1
Host: s3.amazonaws.com
x-amz-date: 20130524T000000Z
x-amz-storage-class: REDUCED_REDUNDANCY
Authorization: SignatureToBeCalculated
x-amz-content-sha256: STREAMING-AWS4-HMAC-SHA256-PAYLOAD
Content-Encoding: aws-chunked
x-amz-decoded-content-length: 66560
Content-Length: 66824
<Payload>
```

The following steps show signature calculations.

1. **Seed signature — Create String to Sign**
   
   a. **CanonicalRequest**

   ```
   PUT /examplebucket/chunkObject.txt
   content-encoding:aws-chunked
   content-length:66824
   host:s3.amazonaws.com
   x-amz-content-sha256:STREAMING-AWS4-HMAC-SHA256-PAYLOAD
   x-amz-date:20130524T000000Z
   x-amz-decoded-content-length:66560
   x-amz-storage-class:REDUCED_REDUNDANCY
   content-encoding;content-length;host;x-amz-content-sha256;x-amz-date;x-amz-decoded-content-length;x-amz-storage-class
   STREAMING-AWS4-HMAC-SHA256-PAYLOAD
   ```

   In the canonical request, the third line is empty because there are no query parameters in the request. The last line is the constant string provided as the value of the hashed Payload, which should be same as the value of x-amz-content-sha256 header.

   b. **StringToSign**

   ```
   AWS4-HMAC-SHA256
   20130524T000000Z
   20130524/us-east-1/s3/aws4_request
   cee3fed04b70f867d036f722359b0bf2f0e5dc0efadbc082b76c4c60e316455
   ```

   **Note**
   For information about each of line in the string to sign, see the diagram that explains seed signature calculation.

2. **SigningKey**

   ```
signing key = HMAC-SHA256(HMAC-SHA256(HMAC-SHA256("AWS4" + "<YourSecretAccessKey>"),"20130524"),"us-east-1"),"s3"),"aws4_request")
   ```
3. **Seed Signature**

| 4f232c4386841ef735655705268965c44a0e4690baa4adea153f7db9fa80a0a9 |

4. **Authorization header**

The resulting Authorization header is as follows:

AWS4-HMAC-SHA256 Credential=AKIAIOSFODNN7EXAMPLE/20130524/us-east-1/s3/aws4_request,SignedHeaders=content-encoding;content-length;host;x-amz-content-sha256;x-amz-date;x-amz-decoded-content-length;x-amz-storage-class,Signature=4f232c4386841ef735655705268965c44a0e4690baa4adea153f7db9fa80a0a9

5. **Chunk 1: (65536 bytes, with value 97 for letter 'a')**

a. Chunk string to sign:

| AWS4-HMAC-SHA256-PAYLOAD 20130524T000000Z 20130524/us-east-1/s3/aws4_request 4f232c4386841ef735655705268965c44a0e4690baa4adea153f7db9fa80a0a9 e3b0c442981f006811f236779df78e19c4415b3427e30d039c940d6aba461303e997645bf718b6f653bebc184e1479f1935b8da974d701b893afcf49e701f3ef9f9c5a |

**Note**

For information about each line in the string to sign, see the preceding diagram that shows various components of the string to sign (for example, the last three lines are, previous-signature, hash(""), and hash(current-chunk-data)).

b. Chunk signature:

| ad80c730a21e5b8d04586a2213dd63b9a0e99e0e2307b0ade35a65485a288648 |

c. Chunk data sent:

| 10000;chunk-signature=ad80c730a21e5b8d04586a2213dd63b9a0e99e0e2307b0ade35a65485a288648 | <65536-bytes> |

6. **Chunk 2: (1024 bytes, with value 97 for letter 'a')**

a. Chunk string to sign:

| AWS4-HMAC-SHA256-PAYLOAD 20130524T000000Z 20130524/us-east-1/s3/aws4_request ad80c730a21e5b8d04586a2213dd63b9a0e99e0e2307b0ade35a65485a288648 e3b0c442981f006811f236779df78e19c4415b3427e30d039c940d6aba461303e997645bf718b6f653bebc184e1479f1935b8da974d701b893afcf49e701f3ef9f9c5a 2edc986847e209b4016e141a6dc87163d207350f416969382d431539bf292e4a |

b. Chunk signature:

| 0055627c9e194cb4542b9e2aa5492e3c1575bbb81b612b7d234b86a503ef5497 |

c. Chunk data sent:

| 400;chunk-signature=0055627c9e194cb4542b9e2aa5492e3c1575bbb81b612b7d234b86a503ef5497 | <1024_bytes> |
7. **Chunk 3: (0 byte data)**

   a. **Chunk string to sign:**

   ```
   AWS4-HMAC-SHA256-PAYLOAD
   20130524/us-east-1/s3/aws4_request
   20130524T000000Z
   e3b0c44298fc1c149afbf4c8996fb92427ae41e4649b933ca445991b7852b855
   e3b0c44298fc1c149afbf4c8996fb92427ae41e4649b933ca445991b7852b855
   ```

   b. **Chunk signature:**

   ```
   b6c6ea8a5354eaf15b3cb7646744f4275b71ea724fed81ceb9323e279d449df9
   ```

   c. **Chunk data sent:**

   ```
   0;chunk-signature=b6c6ea8a5354eaf15b3cb7646744f4275b71ea724fed81ceb9323e279d449df9
   ```

### Authenticating Requests: Using Query Parameters (AWS Signature Version 4)

As described in the authentication overview (see Authentication Methods (p. 734)), you can provide authentication information using query string parameters. Using query parameters to authenticate requests is useful when you want to express a request entirely in a URL. This method is also referred as presigning a URL.

A use case scenario for presigned URLs is that you can grant temporary access to your Amazon S3 resources. For example, you can embed a presigned URL on your website or alternatively use it in command line client (such as Curl) to download objects.

**Note**

You can also use the AWS CLI to create presigned URLs. For more information, see `presign` in the **AWS CLI Command Reference**.

The following is an example presigned URL.

```plaintext
https://s3.amazonaws.com/examplebucket/test.txt
?X-Amz-Algorithm=AWS4-HMAC-SHA256
&X-Amz-Credential=<your-access-key-id>/20130721/us-east-1/s3/aws4_request
&X-Amz-Date=20130721T201207Z
&X-Amz-Expires=86400
&X-Amz-SignedHeaders=host
&X-Amz-Signature=<signature-value>
```

In the example URL, note the following:

- The line feeds are added for readability.
- The `X-Amz-Credential` value in the URL shows the "/" character only for readability. In practice, it should be encoded as `%2F`. For example:

```plaintext
&X-Amz-Credential=<your-access-key-id>%2F20130721%2Fus-east-1%2Fs3%2Faws4_request
```

The following table describes the query parameters in the URL that provide authentication information.
<table>
<thead>
<tr>
<th>Query String Parameter Name</th>
<th>Example Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>X-Amz-Algorithm</td>
<td>Identifies the version of AWS Signature and the algorithm that you used to calculate the signature. For AWS Signature Version 4, you set this parameter value to AWS4-HMAC-SHA256. This string identifies AWS Signature Version 4 (AWS4) and the HMAC-SHA256 algorithm (HMAC-SHA256).</td>
</tr>
<tr>
<td>X-Amz-Credential</td>
<td>In addition to your access key ID, this parameter also provides scope (AWS region and service) for which the signature is valid. This value must match the scope you use in signature calculations, discussed in the following section. The general form for this parameter value is as follows: <code>&lt;your-access-key-id&gt;/&lt;date&gt;/&lt;AWS-region&gt;/&lt;AWS-service&gt;/aws4_request</code> For example: AKIAIOSFODNN7EXAMPLE/20130721/us-east-1/s3/aws4_request For Amazon S3, the AWS-service string is s3. For a list of S3 AWS-region strings, see Regions and Endpoints in the AWS General Reference.</td>
</tr>
<tr>
<td>X-Amz-Date</td>
<td>The date and time format must follow the ISO 8601 standard, and must be formatted with the &quot;yyyyMMdd\THHmmssZ&quot; format. For example if the date and time was &quot;08/01/2016 15:32:41.982-700&quot; then it must first be converted to UTC (Coordinated Universal Time) and then submitted as &quot;20160801T083241Z&quot;.</td>
</tr>
<tr>
<td>X-Amz-Expires</td>
<td>Provides the time period, in seconds, for which the generated presigned URL is valid. For example, 86400 (24 hours). This value is an integer. The minimum value you can set is 1, and the maximum is 604800 (seven days). A presigned URL can be valid for a maximum of seven days because the signing key you use in signature calculation is valid for up to seven days.</td>
</tr>
<tr>
<td>X-Amz-SignedHeaders</td>
<td>Lists the headers that you used to calculate the signature. The following headers are required in the signature calculations: • The HTTP host header. • Any x-amz-* headers that you plan to add to the request. Note For added security, you should sign all the request headers that you plan to include in your request.</td>
</tr>
<tr>
<td>X-Amz-Signature</td>
<td>Provides the signature to authenticate your request. This signature must match the signature Amazon S3 calculates; otherwise, Amazon S3 denies the request. For example, 733255ef022bec3f2a8701cd61d4b371f3f28c9f193a1f02279211d48d5193</td>
</tr>
</tbody>
</table>
Calculating a Signature

The following diagram illustrates the signature calculation process.

The following table describes the functions that are shown in the diagram. You need to implement code for these functions.

<table>
<thead>
<tr>
<th>Function</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Lowercase()</td>
<td>Convert the string to lowercase.</td>
</tr>
<tr>
<td>Hex()</td>
<td>Lowercase base 16 encoding.</td>
</tr>
<tr>
<td>SHA256Hash()</td>
<td>Secure Hash Algorithm (SHA) cryptographic hash function.</td>
</tr>
<tr>
<td>HMAC-SHA256()</td>
<td>Computes HMAC by using the SHA256 algorithm with the signing key provided.</td>
</tr>
<tr>
<td>Trim()</td>
<td>Remove any leading or trailing whitespace.</td>
</tr>
<tr>
<td>UriEncode()</td>
<td>URI encode every byte. UriEncode() must enforce the following rules:</td>
</tr>
<tr>
<td></td>
<td>• URI encode every byte except the unreserved characters: 'A'- 'Z',</td>
</tr>
<tr>
<td></td>
<td>'a'- 'Z', '0'- '9', '+', '-', '/', '_' and '~'.</td>
</tr>
</tbody>
</table>

Signature calculations are described in the following section.

Optional credential parameter if using credentials sourced from the STS service.
Amazon Simple Storage Service API Reference
Calculating a Signature

Function

Description
• The space character is a reserved character and must be encoded
as "%20" (and not as "+").
• Each URI encoded byte is formed by a '%' and the two-digit
hexadecimal value of the byte.
• Letters in the hexadecimal value must be uppercase, for example
"%1A".
• Encode the forward slash character, '/', everywhere except in the
object key name. For example, if the object key name is photos/
Jan/sample.jpg, the forward slash in the key name is not
encoded.

Important

The standard UriEncode functions provided by your
development platform may not work because of
diﬀerences in implementation and related ambiguity
in the underlying RFCs. We recommend that you write
your own custom UriEncode function to ensure that your
encoding will work.
The following is an example UriEncode() function in Java.
public static String UriEncode(CharSequence input, boolean
encodeSlash) {
StringBuilder result = new StringBuilder();
for (int i = 0; i < input.length(); i++) {
char ch = input.charAt(i);
if ((ch >= 'A' && ch <= 'Z') || (ch >= 'a'
&& ch <= 'z') || (ch >= '0' && ch <= '9') || ch == '_' ||
ch == '-' || ch == '~' || ch == '.') {
result.append(ch);
} else if (ch == '/') {
result.append(encodeSlash ? "%2F" : ch);
} else {
result.append(toHexUTF8(ch));
}
}
return result.toString();
}

For more information about the signing process (details of creating a canonical request, string to sign,
and signature calculations), see Signature Calculations for the Authorization Header: Transferring
Payload in a Single Chunk (AWS Signature Version 4) (p. 737). The process is generally the same except
that the creation of CanonicalRequest in a presigned URL diﬀers as follows:
• You don't include a payload hash in the Canonical Request, because when you create a presigned URL,
you don't know the payload content because the URL is used to upload an arbitrary payload. Instead,
you use a constant string UNSIGNED-PAYLOAD.
• The Canonical Query String must include all the query parameters from the preceding table except
for X-Amz-Signature.
• For S3, you must include the X-Amz-Security-Token query parameter in the URL if using
credentials sourced from the STS service.
• Canonical Headers must include the HTTP host header. If you plan to include any of the x-amz-*
headers, these headers must also be added for signature calculation. You can optionally add all other
API Version 2006-03-01
758


headers that you plan to include in your request. For added security, you should sign as many headers as possible.

An Example

Suppose you have an object test.txt in your examplebucket bucket. You want to share this object with others for a period of 24 hours (86400 seconds) by creating a presigned URL.

```
https://s3.amazonaws.com/examplebucket/test.txt
?X-Amz-Algorithm=AWS4-HMAC-SHA256
&X-Amz-Credential=AKIAIOSFODNN7EXAMPLE%2F20130524%2Fus-east-1%2Fs3%2Faws4_request
&X-Amz-Date=20130524T000000Z&X-Amz-Expires=86400&X-Amz-SignedHeaders=host
&X-Amz-Signature=<signature-value>
```

The following steps illustrate first the signature calculations and then construction of the presigned URL. The example makes the following additional assumptions:

- Request timestamp is Fri, 24 May 2013 00:00:00 GMT.
- The bucket is in the US East (N. Virginia) region, and the credential Scope and the Signing Key calculations use us-east-1 as the region specifier. For more information, see Regions and Endpoints in the AWS General Reference.

You can use this example as a test case to verify the signature that your code calculates; however, you must use the same bucket name, object key, time stamp, and the following example credentials:

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>AWSAccessKeyId</td>
<td>AKIAIOSFODNN7EXAMPLE</td>
</tr>
<tr>
<td>AWSSecretAccessKey</td>
<td>wJalrXUtFEMI/K7MDENG/bPxRfiCYEXAMPLEKEY</td>
</tr>
</tbody>
</table>

1. **StringToSign**

   a. **CanonicalRequest**

```
GET
/test.txt
X-Amz-Algorithm=AWS4-HMAC-SHA256&X-Amz-Credential=AKIAIOSFODNN7EXAMPLE%2F20130524%2Fus-east-1%2Fs3%2Faws4_request
&X-Amz-Date=20130524T000000Z&X-Amz-Expires=86400&X-Amz-SignedHeaders=host
host:examplebucket.s3.amazonaws.com
host
UNSIGNED-PAYLOAD
```

   b. **StringToSign**

```
AWS4-HMAC-SHA256
20130524T000000Z
20130524/us-east-1/s3/aws4_request
3bfa292879f6447b2cda7001decf97f4a54dc650c8942174ae0a9121cf58ad04
```
2. **SigningKey**

```plaintext
signing key = HMAC-SHA256(HMAC-SHA256(HMAC-SHA256(HMAC-SHA256("AWS4" + "<YourSecretAccessKey>"), "20130524"), "us-east-1"), "s3"), "aws4_request")
```

3. **Signature**

```plaintext
aeeed9bbccd4d02ee5c0109b86d86835f9953303da4c265957d157751f604d404
```

Now you have all information to construct a presigned URL. The resulting URL for this example is shown as follows (you can use this to compare your presigned URL):

```
https://examplebucket.s3.amazonaws.com/test.txt?X-Amz-Algorithm=AWS4-HMAC-SHA256&X-Amz-Credential=AKIAIOSFODNN7EXAMPLE%2F20130524%2Fus-east-1%2Fs3%2Faws4_request&X-Amz-Date=20130524T000000Z&X-Amz-Expires=86400&X-Amz-SignedHeaders=host&X-Amz-Signature=aeeed9bbccd4d02ee5c0109b86d86835f9953303da4c265957d157751f604d404
```

**Example 2**

The following is an example (unrelated to the previous example) showing a presigned URL with the `X-Amz-Security-Token` parameter.

```
https://examplebucket.s3.us-east-1.amazonaws.com/test.txt?X-Amz-Algorithm=AWS4-HMAC-SHA256&X-Amz-Credential=AKIAIOSFODNN7EXAMPLE%2F20130524%2Fus-east-1%2Fs3%2Faws4_request&X-Amz-Date=20200524T000000Z&X-Amz-Expires=86400&X-Amz-SignedHeaders=host&X-Amz-Security-Token=IQoJb3JpZ2luX2VjEMv%2F%2F%2F%2F%2F%2F%2F%2F%2F%2F%2F%2F%2F%2F%2F%2F%2F8BEAAaDDQ3MjM4NTU0NDY2MCIM83pULBe5%2BNm1GZBKvkBVslSaJVgwsef7f8so2CJ7f756weY13Cw3G5F4hCZcyFpmWYzWhnNh1NnAhnMj5sNkfKLKBx30XAT5P2Gvmq4Vkn9e%2F3hHmkpandQmXn%2B4P98Yv1m%2MrQgSu90jUpGXXE5Ao%2F2608Qo6Qm%3B2ZmaQnansn4YRYrVSEd7072QGZ3%2BvDnDnsso1sYj1v8pF71ujnjvHZRnj0WoeCyMe11L0wT%2F9aUsuHS5E52w%2FyUoCOn00aZuyROuVzYVQ70ebWPhVWhVYUT%2B0UHkmun6vpo4bA2N2Qgjbjtb3fI92v%2Bx5kY160s8QE866jDjTUmMPSldMzIC1GZ3p0mN5dzeYLM3GojI%2F01mWkPQw6gm3tSp0a9e0eumAPTA7qMgy9HNUTKBDsX9E627wVzBum35J7hExxV48euaadKgrIRX3279ruQF5Qoc9LUJrDjR%2B480kAagjX19GX8Q8V8JaLxghqEmwd6U4XcW7ku3gaMBzqTf7NDX%3D&X-Amz-Signature=<signature-value>
```

### Examples: Signature Calculations in AWS Signature Version 4

**Topics**

- Signature Calculation Examples Using Java (AWS Signature Version 4) (p. 761)
- Examples of Signature Calculations Using C# (AWS Signature Version 4) (p. 762)

For authenticated requests, unless you are using the AWS SDKs, you have to write code to calculate signatures that provide authentication information in your requests. Signature calculation in AWS Signature Version 4 (see Authenticating Requests (AWS Signature Version 4) (p. 733)) can be a complex undertaking, and we recommend that you use the AWS SDKs whenever possible.
This section provides examples of signature calculations written in Java and C#. The code samples send the following requests and use the HTTP Authorization header to provide authentication information:

- **PUT object** – Separate examples illustrate both uploading the full payload at once and uploading the payload in chunks. For information about using the Authorization header for authentication, see Authenticating Requests: Using the Authorization Header (AWS Signature Version 4) (p. 735).

- **GET object** – This example generates a presigned URL to get an object. Query parameters provide the signature and other authentication information. Users can paste a presigned URL in their browser to retrieve the object, or you can use the URL to create a clickable link. For information about using query parameters for authentication, see Authenticating Requests: Using Query Parameters (AWS Signature Version 4) (p. 755).

The rest of this section describes the examples in Java and C#. The topics include instructions for downloading the samples and for executing them.

**Signature Calculation Examples Using Java (AWS Signature Version 4)**

The Java sample that shows signature calculation can be downloaded at https://docs.aws.amazon.com/AmazonS3/latest/API/samples/AWSS3SigV4JavaSamples.zip. In RunAllSamples.java, the main() function executes sample requests to create an object, retrieve an object, and create a presigned URL for the object. The sample creates an object from the text string provided in the code:

```java
PutS3ObjectSample.putS3Object(bucketName, regionName, awsAccessKey, awsSecretKey);
GetS3ObjectSample.getS3Object(bucketName, regionName, awsAccessKey, awsSecretKey);
PresignedUrlSample.getPresignedUrlToS3Object(bucketName, regionName, awsAccessKey, awsSecretKey);
PutS3ObjectChunkedSample.putS3ObjectChunked(bucketName, regionName, awsAccessKey, awsSecretKey);
```

To test the examples on a Linux-based computer

The following instructions are for the Linux operating system.

1. In a terminal, navigate to the directory that contains AWSS3SigV4JavaSamples.zip.
2. Extract the .zip file.
3. In a text editor, open the file ./com/amazonaws/services/s3/samples/RunAllSamples.java. Update code with the following information:
   - The name of a bucket where the new object can be created.
   - AWS region where the bucket resides.

   **Note**
   The examples use a virtual-hosted style request to access the bucket. To avoid potential errors, ensure that your bucket name conforms to the bucket naming rules as explained in Bucket Restrictions and Limitations in the Amazon Simple Storage Service Developer Guide.

4. Compile the source code and store the compiled classes into the bin/ directory.

   `javac -d bin -source 6 -verbose com`

5. Change the directory to bin/, and then run RunAllSamples.
java com.amazonaws.services.s3.sample.RunAllSamples

The code runs all the methods in main(). For each request, the output will show the canonical request, the string to sign, and the signature.

Examples of Signature Calculations Using C# (AWS Signature Version 4)

The C# sample that shows signature calculation can be downloaded at https://docs.aws.amazon.com/AmazonS3/latest/API/samples/AmazonS3SigV4_Samples_CSharp.zip. In Program.cs, the main() function executes sample requests to create an object, retrieve an object, and create a presigned URL for the object. The code for signature calculation is in the Signers folder.

```csharp
PutS3ObjectSample.Run(awsRegion, bucketName, "MySampleFile.txt");
Console.WriteLine("\n
************************************************");
PutS3ObjectChunkedSample.Run(awsRegion, bucketName, "MySampleFileChunked.txt");
Console.WriteLine("\n
************************************************");
GetS3ObjectSample.Run(awsRegion, bucketName, "MySampleFile.txt");
Console.WriteLine("\n
************************************************");
PresignedUrlSample.Run(awsRegion bucketName, "MySampleFile.txt");
```

To test the examples with Microsoft Visual Studio 2010 or later

1. Extract the .zip file.
2. Start Visual Studio, and then open the .sln file.
3. Update the App.config file with valid security credentials.
4. Update the code as follows:
   • In Program.cs, provide the bucket name and the AWS region where the bucket resides. The sample creates an object in this bucket.
5. Run the code.
6. To verify that the object was created, copy the presigned URL that the program creates, and then paste it in a browser window.

Authenticating Requests: Browser-Based Uploads Using POST (AWS Signature Version 4)

Amazon S3 supports HTTP POST requests so that users can upload content directly to Amazon S3. Using HTTP POST to upload content simplifies uploads and reduces upload latency where users upload data to store in Amazon S3. This section describes how you authenticate HTTP POST requests. For more information about HTTP POST requests, how to create a form, create a POST policy, and an example, see Authenticating Requests in Browser-Based Uploads Using POST (AWS Signature Version 4) (p. 768).

To authenticate an HTTP POST request you do the following:

1. The form must include the following fields to provide signature and relevant information that Amazon S3 can use to re-calculate the signature upon receiving the request:
### Authenticating HTTP POST Requests

<table>
<thead>
<tr>
<th>Element Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>policy</td>
<td>The Base64-encoded security policy that describes what is permitted in the request. For signature calculation this policy is the string you sign. Amazon S3 must get this policy so it can re-calculate the signature.</td>
</tr>
<tr>
<td>x-amz-algorithm</td>
<td>The signing algorithm used. For AWS Signature Version 4, the value is <code>AWS4-HMAC-SHA256</code>.</td>
</tr>
<tr>
<td>x-amz-credential</td>
<td>In addition to your access key ID, this provides scope information you used in calculating the signing key for signature calculation.</td>
</tr>
<tr>
<td></td>
<td>It is a string of the following form:</td>
</tr>
<tr>
<td></td>
<td><code>&lt;your-access-key-id&gt;/&lt;date&gt;/&lt;aws-region&gt;/&lt;aws-service&gt;/aws4_request</code></td>
</tr>
<tr>
<td></td>
<td>For example:</td>
</tr>
<tr>
<td></td>
<td>AKIAIOSFODNN7EXAMPLE/20130728/us-east-1/s3/aws4_request.</td>
</tr>
<tr>
<td></td>
<td>For Amazon S3, the <code>aws-service</code> string is <code>s3</code>. For a list of Amazon S3 <code>aws-region</code> strings, see Regions and Endpoints in the AWS General Reference.</td>
</tr>
<tr>
<td>x-amz-date</td>
<td>It is the date value in ISO8601 format. For example, 20130728T000000Z.</td>
</tr>
<tr>
<td></td>
<td>It is the same date you used in creating the signing key. This must also be the same value you provide in the policy (x-amz-date) that you signed.</td>
</tr>
<tr>
<td>x-amz-signature</td>
<td>(AWS Signature Version 4) The HMAC-SHA256 hash of the security policy.</td>
</tr>
<tr>
<td></td>
<td>For more information on options for the signature, see Add the signature to the HTTP request in the AWS General Reference.</td>
</tr>
</tbody>
</table>

2. The POST policy must include the following elements:

<table>
<thead>
<tr>
<th>Element Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>x-amz-algorithm</td>
<td>The signing algorithm that you used to calculation the signature. For AWS Signature Version 4, the value is <code>AWS4-HMAC-SHA256</code>.</td>
</tr>
<tr>
<td>x-amz-credential</td>
<td>In addition to your access key ID, this provides scope information you used in calculating the signing key for signature calculation.</td>
</tr>
<tr>
<td></td>
<td>It is a string of the following form:</td>
</tr>
<tr>
<td></td>
<td><code>&lt;your-access-key-id&gt;/&lt;date&gt;/&lt;aws-region&gt;/&lt;aws-service&gt;/aws4_request</code></td>
</tr>
</tbody>
</table>
### Calculating a Signature

The following diagram illustrates the signature calculation process.

#### To Calculate a Signature

1. Create a policy using UTF-8 encoding.
2. Convert the UTF-8-encoded policy to Base64. The result is the string to sign.
3. Create the signature as an HMAC-SHA256 hash of the string to sign. You will provide the signing key as key to the hash function.
4. Encode the signature by using hex encoding.

For more information about creating HTML forms, security policies, and an example, see the following subtopics:

- Creating an HTML Form (Using AWS Signature Version 4) (p. 797)
- Creating a POST Policy (p. 801)
- Example: Browser-Based Upload using HTTP POST (Using AWS Signature Version 4) (p. 807)
- Using POST with Adobe Flash to Upload Objects (p. 809)
Amazon S3 Signature Version 4 Authentication
Specific Policy Keys

The following table shows the policy keys related Amazon S3 Signature Version 4 authentication that can be in Amazon S3 policies. In a bucket policy, you can add these conditions to enforce specific behavior when requests are authenticated by using Signature Version 4. For example policies, see Bucket Policy Examples Using Signature Version 4 Related Condition Keys (p. 766).

Applicable Keys for s3:* Actions or any of the Amazon S3 Actions

<table>
<thead>
<tr>
<th>Applicable Keys</th>
<th>Description</th>
</tr>
</thead>
</table>
| s3:signatureversion | Identifies the version of AWS Signature that you want to support for authenticated requests. For authenticated requests, Amazon S3 supports both Signature Version 4 and Signature Version 2. You can add this condition in your bucket policy to require a specific signature version.  
Valid values:  
"AWS" identifies Signature Version 2  
"AWS4-HMAC-SHA256" identifies Signature Version 4 |
| s3:authType       | Amazon S3 supports various methods of authentication (see Authenticating Requests (AWS Signature Version 4) (p. 733). You can optionally use this condition key to restrict incoming requests to use a specific authentication method. For example, you can allow only the HTTP Authorization header to be used in request authentication.  
Valid values:  
REST-HEADER  
REST-QUERY-STRING  
POST |
| s3:signatureAge   | The length of time, in milliseconds, that a signature is valid in an authenticated request.  
This condition works only for presigned URLs (the most restrictive condition wins).  
In Signature Version 4, the signing key is valid for up to seven days (see Introduction to Signing Requests (p. 734). Therefore, the signatures are also valid for up to seven days. You can use this condition to further limit the signature age.  
Example value: 100 |
## Applicable Keys

<table>
<thead>
<tr>
<th>Key</th>
<th>Description</th>
</tr>
</thead>
</table>
| s3:x-amz-content-sha256                  | You can use this condition key to disallow unsigned content in your bucket. When you use Signature Version 4, for requests that use the Authorization header, you add the x-amz-content-sha256 header in the signature calculation and then set its value to the hash payload. You can use this condition key in your bucket policy to deny any uploads where payloads are not signed. For example:  
  - Deny uploads that use presigned URLs. For more information, see [Authenticating Requests: Using Query Parameters (AWS Signature Version 4)](p. 755).  
  - Deny uploads that use Authorization header to authenticate requests but don't sign the payload. For more information, see [Signature Calculations for the Authorization Header: Transferring Payload in a Single Chunk (AWS Signature Version 4)](p. 737).  
  Valid value: UNSIGNED-PAYLOAD |

## Bucket Policy Examples Using Signature Version 4 Related Condition Keys

Deny any Amazon S3 action on the examplebucket to anyone if request is authenticated using Signature Version 4.

```json
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "Test",
      "Effect": "Deny",
      "Principal": "*",
      "Action": "s3:*",
      "Resource": "arn:aws:s3:::examplebucket/*",
      "Condition": {
        "StringEquals": {
          "s3:signatureversion": "AWS4-HMAC-SHA256"
        }
      }
    }
  ]
}
```

The following bucket policy denies any Amazon S3 presigned URL request on objects in examplebucket if the signature is more than ten minutes old.
The following bucket policy allows only requests that use the Authorization header for request authentication. Any POST or presigned URL requests will be denied.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "Allow only requests that use Authorization header for request authentication. Deny POST or presigned URL requests."
        }
    ]
}
```

The following bucket policy denies any uploads with unsigned payloads, such as uploads using presigned URLs.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "Deny uploads with unsigned payloads."
        }
    ]
}
```
Authenticating Requests in Browser-Based Uploads Using POST (AWS Signature Version 4)

This section discusses how to upload files directly to Amazon S3 through a browser using HTTP POST requests. It also contains information about how to use the AWS Amplify JavaScript library for browser-based file uploads to Amazon S3.

Topics
- POST Object (p. 769)
- POST Object restore (p. 781)
- Browser-Based Uploads Using HTTP POST (p. 795)
- Calculating a Signature (p. 796)
- Creating an HTML Form (Using AWS Signature Version 4) (p. 797)
- Creating a POST Policy (p. 801)
- Example: Browser-Based Upload using HTTP POST (Using AWS Signature Version 4) (p. 807)
- Using POST with Adobe Flash to Upload Objects (p. 809)
- Browser-Based Uploads to Amazon S3 Using the AWS Amplify Library (p. 810)
POST Object

Description

The POST operation adds an object to a specified bucket using HTML forms. POST is an alternate form of PUT that enables browser-based uploads as a way of putting objects in buckets. Parameters that are passed to PUT via HTTP Headers are instead passed as form fields to POST in the multipart/form-data encoded message body. You must have WRITE access on a bucket to add an object to it. Amazon S3 never stores partial objects: if you receive a successful response, you can be confident the entire object was stored.

Amazon S3 is a distributed system. If Amazon S3 receives multiple write requests for the same object simultaneously, all but the last object written is overwritten.

To ensure that data is not corrupted traversing the network, use the Content-MD5 form field. When you use this form field, Amazon S3 checks the object against the provided MD5 value. If they do not match, Amazon S3 returns an error. Additionally, you can calculate the MD5 value while posting an object to Amazon S3 and compare the returned ETag to the calculated MD5 value. The ETag only reflects changes to the contents of an object, not its metadata.

Note
To configure your application to send the Request Headers before sending the request body, use the 100-continue HTTP status code. For POST operations, this helps you avoid sending the message body if the message is rejected based on the headers (for example, authentication failure or redirect). For more information on the 100-continue HTTP status code, go to Section 8.2.3 of http://www.ietf.org/rfc/rfc2616.txt.

You can optionally request server-side encryption where Amazon S3 encrypts your data as it writes it to disks in its data centers and decrypts it for you when you access it. You have the option of providing your own encryption key or you can use the AWS-managed encryption keys. For more information, go to Using Server-Side Encryption in the Amazon Simple Storage Service Developer Guide.

Versioning

If you enable versioning for a bucket, POST automatically generates a unique version ID for the object being added. Amazon S3 returns this ID in the response using the x-amz-version-id response header.

If you suspend versioning for a bucket, Amazon S3 always uses null as the version ID of the object stored in a bucket.

For more information about returning the versioning state of a bucket, see GET Bucket (Versioning Status) (p. 148).

Amazon S3 is a distributed system. If you enable versioning for a bucket and Amazon S3 receives multiple write requests for the same object simultaneously, all of the objects are stored.

To see sample requests that use versioning, see Sample Request (p. 779).

Requests

Syntax

```
POST / HTTP/1.1
Host: destinationBucket.s3.amazonaws.com
User-Agent: browser_data
```
Accept: file_types
Accept-Language: Regions
Accept-Encoding: encoding
Accept-Charset: character_set
Keep-Alive: 300
Connection: keep-alive
Content-Type: multipart/form-data; boundary=9431149156168
Content-Length: length

--9431149156168
Content-Disposition: form-data; name="key"

acl
--9431149156168
Content-Disposition: form-data; name="tagging"
<Tagging><TagSet><Tag><Key>Tag Name</Key><Value>Tag Value</Value></Tag></TagSet></Tagging>

--9431149156168
Content-Disposition: form-data; name="success_action_redirect"

success_redirect
--9431149156168
Content-Disposition: form-data; name="Content-Type"

content_type
--9431149156168
Content-Disposition: form-data; name="x-amz-meta-uuid"

uuid
--9431149156168
Content-Disposition: form-data; name="x-amz-meta-tag"

metadata
--9431149156168
Content-Disposition: form-data; name="AWSAccessKeyId"

access-key-id
--9431149156168
Content-Disposition: form-data; name="Policy"

encoded_policy
--9431149156168
Content-Disposition: form-data; name="Signature"

signature=
--9431149156168
Content-Disposition: form-data; name="file"; filename="MyFilename.jpg"
Content-Type: image/jpeg

file_content
--9431149156168
Content-Disposition: form-data; name="submit"

Upload to Amazon S3
--9431149156168--

**Request Parameters**

This implementation of the operation does not use request parameters.

**Form Fields**

This operation can use the following form fields.
<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
<th>Required</th>
</tr>
</thead>
<tbody>
<tr>
<td>AWSAccessKeyId</td>
<td>The AWS access key ID of the owner of the bucket who grants an Anonymous user access for a request that satisfies the set of constraints in the policy.</td>
<td>Conditional</td>
</tr>
<tr>
<td>acl</td>
<td>Specifies an Amazon S3 access control list. If an invalid access control list is specified, an error is generated. For more information on ACLs, go to Access Control List (ACL) Overview in the Amazon Simple Storage Service Developer Guide.</td>
<td>No</td>
</tr>
<tr>
<td>Cache-Control, Content-Type, Content-Disposition, Content-Encoding, Expires</td>
<td>REST-specific headers. For more information, see PutObject (p. 343).</td>
<td>No</td>
</tr>
<tr>
<td>file</td>
<td>File or text content.</td>
<td>Yes</td>
</tr>
<tr>
<td>key</td>
<td>The name of the uploaded key.</td>
<td>Yes</td>
</tr>
</tbody>
</table>

**Type:** File or text content.

The file or text content must be the last field in the form.

You cannot upload more than one file at a time.
<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
<th>Required</th>
</tr>
</thead>
</table>
| policy                | Security Policy describing what is permitted in the request. Requests without a security policy are considered anonymous and work only on publicly writable buckets. For more information, go to HTML Forms and Upload Examples.  
Type: String  
Default: None  
Constraints: Policy is required if the bucket is not publicly writable. | Conditional|
| success_action_redirect, redirect | The URL to which the client is redirected upon successful upload.  
If `success_action_redirect` is not specified, Amazon S3 returns the empty document type specified in the `success_action_status` field.  
If Amazon S3 cannot interpret the URL, it acts as if the field is not present.  
If the upload fails, Amazon S3 displays an error and does not redirect the user to a URL.  
Type: String  
Default: None  
**Note**  The redirect field name is deprecated, and support for the redirect field name is removed in the future. | No         |
<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
<th>Required</th>
</tr>
</thead>
<tbody>
<tr>
<td>success_action_status</td>
<td>If you don't specify <code>success_action_redirect</code>, the status code is returned to the client when the upload succeeds.</td>
<td>No</td>
</tr>
<tr>
<td></td>
<td>Accepts the values 200, 201, or 204 (the default).</td>
<td></td>
</tr>
<tr>
<td></td>
<td>If the value is set to 200 or 204, Amazon S3 returns an empty document with a 200 or 204 status code.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>If the value is set to 201, Amazon S3 returns an XML document with a 201 status code.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>If the value is not set or if it is set to an invalid value, Amazon S3 returns an empty document with a 204 status code.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Type: String</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Default: None</td>
<td></td>
</tr>
<tr>
<td></td>
<td><strong>Note</strong></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Some versions of the Adobe Flash player do not properly handle HTTP responses with an empty body. To support uploads through Adobe Flash, we recommend setting <code>success_action_status</code> to 201.</td>
<td></td>
</tr>
<tr>
<td>tagging</td>
<td>Specifies set of tags to add to the object using the following encoding scheme.</td>
<td>No</td>
</tr>
<tr>
<td></td>
<td><code>&lt;Tagging&gt;</code></td>
<td></td>
</tr>
<tr>
<td></td>
<td><code>&lt;TagSet&gt;</code></td>
<td></td>
</tr>
<tr>
<td></td>
<td><code>&lt;Tag&gt;</code></td>
<td></td>
</tr>
<tr>
<td></td>
<td><code>&lt;Key&gt;</code>Tag Name&lt;/Key&gt;`</td>
<td></td>
</tr>
<tr>
<td></td>
<td><code>&lt;Value&gt;</code>Tag Value&lt;/Value&gt;`</td>
<td></td>
</tr>
<tr>
<td></td>
<td><code>&lt;/Tag&gt;</code></td>
<td></td>
</tr>
<tr>
<td></td>
<td><code>...</code></td>
<td></td>
</tr>
<tr>
<td></td>
<td><code>&lt;/TagSet&gt;</code></td>
<td></td>
</tr>
<tr>
<td></td>
<td><code>&lt;/Tagging&gt;</code></td>
<td></td>
</tr>
<tr>
<td></td>
<td><strong>For more information, see</strong> <a href="http://docs.aws.amazon.com/AmazonS3/latest/dev/ObjectTagging.html">Object Tagging in the Amazon Simple Storage Service Developer Guide</a>.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Type: String</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Default: None</td>
<td></td>
</tr>
<tr>
<td>Name</td>
<td>Description</td>
<td>Required</td>
</tr>
<tr>
<td>-----------------------</td>
<td>-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
<td>----------</td>
</tr>
<tr>
<td>x-amz-storage-class</td>
<td>Storage class to use for storing the object. If you don't specify a class, Amazon S3 uses the default storage class, STANDARD. Amazon S3 supports other storage classes. For more information, see Storage Classes in the Amazon Simple Storage Service Developer Guide.</td>
<td>No</td>
</tr>
<tr>
<td></td>
<td>Type: String</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Default: STANDARD</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Valid values: STANDARD</td>
<td>REDUCED_REDUNDANCY</td>
</tr>
<tr>
<td>x-amz-meta-*</td>
<td>Headers starting with this prefix are user-defined metadata. Each one is stored and returned as a set of key-value pairs. Amazon S3 doesn't validate or interpret user-defined metadata. For more information, see PutObject (p. 343).</td>
<td>No</td>
</tr>
<tr>
<td></td>
<td>Type: String</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Default: None</td>
<td></td>
</tr>
<tr>
<td>x-amz-security-token</td>
<td>Amazon DevPay security token. Each request that uses Amazon DevPay requires two x-amz-security-token form fields: one for the product token and one for the user token.</td>
<td>No</td>
</tr>
<tr>
<td></td>
<td>Type: String</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Default: None</td>
<td></td>
</tr>
</tbody>
</table>
### Server-Side Encryption Specific Request Form Fields

You can optionally request Amazon S3 to encrypt data at rest using server-side encryption. Server-side encryption is data encryption at rest. Amazon S3 encrypts your data as it writes it to disks in its data centers and decrypts it when you access it.

For more information, see Protecting Data Using Server-Side Encryption in the Amazon Simple Storage Service Developer Guide.

Depending on whether you want to use AWS-managed encryption keys or provide your own encryption keys, the following form fields:

- Use AWS-managed encryption keys — If you want Amazon S3 to manage keys used to encrypt data, specify the following form fields in the request.

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
<th>Required</th>
</tr>
</thead>
<tbody>
<tr>
<td>x-amz-server-side-encryption</td>
<td>Specifies a server-side encryption algorithm to use when Amazon S3 creates an object.</td>
<td>Yes</td>
</tr>
</tbody>
</table>
### Name | Description | Required
--- | --- | ---
Valid Value: `aws:kms`, AES256 | |  
**`x-amz-server-side-encryption-aws-kms-key-id`** | If the `x-amz-server-side-encryption` is present and has the value of `aws:kms`, this header specifies the ID of the AWS Key Management Service (AWS KMS) master encryption key that was used for the object.  
Type: String | Yes, if the value of `x-amz-server-side-encryption` is `aws:kms`  

**`x-amz-server-side-encryption-context`** | If `x-amz-server-side-encryption` is present, and if its value is `aws:kms`, this header specifies the encryption context for the object. The value of this header is a base64-encoded UTF-8 string holding JSON with the key-value pairs for the encryption context.  
Type: String | No  

**Note**  
If you specify `x-amz-server-side-encryption:aws:kms`, but do not provide `x-amz-server-side-encryption-aws-kms-key-id`, Amazon S3 uses the default AWS KMS key to protect the data.  

- Use customer-provided encryption keys — If you want to manage your own encryption keys, you must provide all the following form fields in the request.  

**Note**  
If you use this feature, the ETag value that Amazon S3 returns in the response is not the MD5 of the object.  

### Name | Description | Required
--- | --- | ---
**`x-amz-server-side-encryption-customer-algorithm`** | Specifies the algorithm to use when encrypting the object.  
Type: String  
Default: None  
Valid Value: AES256  
Constraints: Must be accompanied by valid `x-amz-server-side-encryption-customer-key` and `x-amz-server-side-encryption-customer-key-MD5` fields. | Yes  

**`x-amz-server-side-encryption-customer-key`** | Specifies the customer-provided base64-encoded encryption key for Amazon S3 to use in encrypting data. This value is used to store the object and then it is discarded. Amazon does not store the encryption key. The key must be appropriate for use with the algorithm specified in the `x-amz-server-side-encryption-customer-algorithm` header.  
Type: String  
Default: None | Yes
### Requests

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
<th>Required</th>
</tr>
</thead>
<tbody>
<tr>
<td>x-amz-server-side-encryption-</td>
<td>Specifies the base64-encoded 128-bit MD5 digest of the encryption key according to RFC 1321. Amazon S3 uses this header for a message integrity check to ensure that the encryption key was transmitted without error.</td>
<td>Yes</td>
</tr>
<tr>
<td>customer-key-MD5</td>
<td>Type: String</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Default: None</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Constraints: Must be accompanied by valid x-amz-server-side-encryption-customer-algorithm and x-amz-server-side-encryption-customer-key-MD5 fields.</td>
<td></td>
</tr>
</tbody>
</table>

#### Responses

**Response Headers**

This implementation of the operation can include the following response headers in addition to the response headers common to all responses. For more information, see Common Response Headers (p. 814).

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>x-amz-expiration</td>
<td>If an Expiration action is configured for the object as part of the bucket's lifecycle configuration, Amazon S3 returns this header. The header value includes an &quot;expiry-date&quot; component and a URL-encoded &quot;rule-id&quot; component. For version-enabled buckets, this header applies only to current versions. Amazon S3 does not provide a header to infer when a noncurrent version is eligible for permanent deletion. For more information, see PutBucketLifecycleConfiguration (p. 292).</td>
<td></td>
</tr>
<tr>
<td>success_action_redirect, redirect</td>
<td>The URL to which the client is redirected on successful upload.</td>
<td></td>
</tr>
<tr>
<td>x-amz-server-side-encryption</td>
<td>If you specified server-side encryption either with AWS KMS encryption or AWS-managed encryption in your POST request, the response includes this header. It confirms the encryption algorithm that Amazon S3 used to encrypt the object.</td>
<td></td>
</tr>
</tbody>
</table>

API Version 2006-03-01

777
### Name | Description
---|---
x-amz-server-side-encryption-aws-kms-key-id | If the x-amz-server-side-encryption header is present and has the value of aws:kms, this header specifies the ID of the AWS KMS master encryption key that was used for the object. 
Type: String

x-amz-server-side-encryption-customer-algorithm | If server-side encryption with customer-provided encryption keys (SSE-C) encryption was requested, the response includes this header that confirms the encryption algorithm that was used. 
Type: String
Valid Values: AES256

x-amz-server-side-encryption-customer-key-MD5 | If SSE-C encryption was requested, the response includes this header to verify roundtrip message integrity of the customer-provided encryption key. 
Type: String

x-amz-version-id | Version of the object. 
Type: String

### Response Elements

### Name | Description
---|---
Bucket | Name of the bucket the object was stored in. 
Type: String 
Ancestor: PostResponse

ETag | The entity tag is an MD5 hash of the object that you can use to do conditional GET operations using the If-Modified request tag with the GET request operation. ETag reflects changes only to the contents of an object, not its metadata. 
Type: String 
Ancestor: PostResponse

Key | The object key name. 
Type: String 
Ancestor: PostResponse

Location | URI of the object. 
Type: String 
Ancestor: PostResponse
Special Errors

This implementation of the operation does not return special errors. For general information about Amazon S3 errors and a list of error codes, see Error Responses (p. 816).

Examples

Sample Request

```plaintext
POST /Neo HTTP/1.1
Content-Length: 4
Host: quotes.s3.amazonaws.com
Date: Wed, 01 Mar 2006 12:00:00 GMT
Authorization: authorization string
Content-Type: text/plain
Expect: the 100-continue HTTP status code
```

Sample Response with Versioning Suspended

The following is a sample response when bucket versioning is suspended:

```plaintext
HTTP/1.1 100 Continue
HTTP/1.1 200 OK
x-amz-id-2: LriYPLdMoAIfgSmF/1YsViT1LW94/xUQxMsF7xiEblA0wI0Ixl+zbwZ163pt7
x-amz-request-id: 0A49CE406975EAC
x-amz-version-id: default
Date: Wed, 12 Oct 2009 17:50:00 GMT
ETag: "1b2cf535f27731c974343645a3985328"
Content-Length: 0
Connection: close
Server: AmazonS3
```

In this response, the version ID is null.

Sample Response with Versioning Enabled

The following is a sample response when bucket versioning is enabled.

```plaintext
HTTP/1.1 100 Continue
HTTP/1.1 200 OK
x-amz-id-2: LriYPLdMoAIfgSmF/1YsViT1LW94/xUQxMsF7xiEblA0wI0Ixl+zbwZ163pt7
x-amz-request-id: 0A49CE406975EAC
x-amz-version-id: 43jfkodU8493jnFJD9fjj3HhNVfdsQUIFDnMsf038jfdsjGFDSIRp
Date: Wed, 01 Mar 2006 12:00:00 GMT
ETag: "828ef3f7f96f00ad9f27c383fc9ac7f"
Content-Length: 0
Connection: close
Server: AmazonS3
```

Related Resources

- CopyObject (p. 17)
- POST Object (p. 769)
- GetObject (p. 154)
POST Object restore

Description

This operation performs the following types of requests:

- **select** – Perform a select query on an archived object
- **restore an archive** – Restore an archived object

To use this operation, you must have permissions to perform the `s3:RestoreObject` and `s3:GetObject` actions. The bucket owner has this permission by default and can grant this permission to others. For more information about permissions, see Permissions Related to Bucket Subresource Operations and Managing Access Permissions to Your Amazon S3 Resources in the Amazon Simple Storage Service Developer Guide.

Querying Archives with Select Requests

You use a select type of request to perform SQL queries on archived objects. The archived objects that are being queried by the select request must be formatted as uncompressed comma-separated values (CSV) files. You can run queries and custom analytics on your archived data without having to restore your data to a hotter Amazon S3 tier. For an overview about select requests, see Querying Archived Objects in the Amazon Simple Storage Service Developer Guide.

When making a select request, do the following:

- Define an output location for the select query's output. This must be an Amazon S3 bucket in the same AWS Region as the bucket that contains the archive object that is being queried. The AWS account that initiates the job must have permissions to write to the S3 bucket. You can specify the storage class and encryption for the output objects stored in the bucket. For more information about output, see Querying Archived Objects in the Amazon Simple Storage Service Developer Guide.

For more information about the S3 structure in the request body, see the following:

- **PutObject** (p. 343)
- Managing Access with ACLs in the Amazon Simple Storage Service Developer Guide
- Protecting Data Using Server-Side Encryption in the Amazon Simple Storage Service Developer Guide
- Define the SQL expression for the `SELECT` type of restoration for your query in the request body's `SelectParameters` structure. You can use expressions like the following examples.
- The following expression returns all records from the specified object.

```
SELECT * FROM Object
```

- Assuming that you are not using any headers for data stored in the object, you can specify columns with positional headers.

```
SELECT s._1, s._2 FROM Object s WHERE s._3 > 100
```

- If you have headers and you set the `fileHeaderInfo` in the CSV structure in the request body to `USE`, you can specify headers in the query. (If you set the `fileHeaderInfo` field to `IGNORE`, the first row is skipped for the query.) You cannot mix ordinal positions with header column names.

```
SELECT s.Id, s.FirstName, s.SSN FROM S3Object s
```
For more information about using SQL with S3 Glacier Select restore, see SQL Reference for Amazon S3 Select and S3 Glacier Select in the Amazon Simple Storage Service Developer Guide.

When making a select request, you can also do the following:

- To expedite your queries, specify the Expedited tier. For more information about tiers, see "Restoring Archives," later in this topic.
- Specify details about the data serialization format of both the input object that is being queried and the serialization of the CSV-encoded query results.

The following are additional important facts about the select feature:

- The output results are new Amazon S3 objects. Unlike archive retrievals, they are stored until explicitly deleted—manually or through a lifecycle policy.
- You can issue more than one select request on the same Amazon S3 object. Amazon S3 doesn’t deduplicate requests, so avoid issuing duplicate requests.
- Amazon S3 accepts a select request even if the object has already been restored. A select request doesn't return error response 409.

Restoring Archives

Objects in the GLACIER and DEEP_ARCHIVE storage classes are archived. To access an archived object, you must first initiate a restore request. This restores a temporary copy of the archived object. In a restore request, you specify the number of days that you want the restored copy to exist. After the specified period, Amazon S3 deletes the temporary copy but the object remains archived in the GLACIER or DEEP_ARCHIVE storage class that object was restored from.

To restore a specific object version, you can provide a version ID. If you don't provide a version ID, Amazon S3 restores the current version.

The time it takes restore jobs to finish depends on which storage class the object is being restored from and which data access tier you specify.

When restoring an archived object (or using a select request), you can specify one of the following data access tier options in the Tier element of the request body:

- **Expedited** - Expedited retrievals allow you to quickly access your data stored in the GLACIER storage class when occasional urgent requests for a subset of archives are required. For all but the largest archived objects (250 MB+), data accessed using Expedited retrievals are typically made available within 1–5 minutes. Provisioned capacity ensures that retrieval capacity for Expedited retrievals is available when you need it. Expedited retrievals and provisioned capacity are not available for the DEEP_ARCHIVE storage class.

- **Standard** - Standard retrievals allow you to access any of your archived objects within several hours. This is the default option for the GLACIER and DEEP_ARCHIVE retrieval requests that do not specify the retrieval option. Standard retrievals typically complete within 3–5 hours from the GLACIER storage class and typically complete within 12 hours from the DEEP_ARCHIVE storage class.

- **Bulk** - Bulk retrievals are Amazon S3 Glacier's lowest-cost retrieval option, enabling you to retrieve large amounts, even petabytes, of data inexpensively in a day. Bulk retrievals typically complete within 5–12 hours from the GLACIER storage class and typically complete within 48 hours from the DEEP_ARCHIVE storage class.

For more information about archive retrieval options and provisioned capacity for Expedited data access, see Restoring Archived Objects in the Amazon Simple Storage Service Developer Guide.
You can use Amazon S3 restore speed upgrade to change the restore speed to a faster speed while it is in progress. You upgrade the speed of an in-progress restoration by issuing another restore request to the same object, setting a new `Tier` request element. When issuing a request to upgrade the restore tier, you must choose a tier that is faster than the tier that the in-progress restore is using. You must not change any other parameters, such as the `Days` request element. For more information, see Upgrading the Speed of an In-Progress Restore in the Amazon Simple Storage Service Developer Guide.

To get the status of object restoration, you can send a HEAD request. Operations return the `x-amz-restore` header, which provides information about the restoration status, in the response. You can use Amazon S3 event notifications to notify you when a restore is initiated or completed. For more information, see Configuring Amazon S3 Event Notifications in the Amazon Simple Storage Service Developer Guide.

After restoring an archived object, you can update the restoration period by reissuing the request with a new period. Amazon S3 updates the restoration period relative to the current time and charges only for the request—there are no data transfer charges. You cannot update the restoration period when Amazon S3 is actively processing your current restore request for the object.

If your bucket has a lifecycle configuration with a rule that includes an expiration action, the object expiration overrides the life span that you specify in a restore request. For example, if you restore an object copy for 10 days, but the object is scheduled to expire in 3 days, Amazon S3 deletes the object in 3 days. For more information about lifecycle configuration, see PutBucketLifecycleConfiguration (p. 292) and Object Lifecycle Management in Amazon Simple Storage Service Developer Guide.

**Requests**

**Syntax**

```
POST /ObjectName?restore&versionId=VersionID HTTP/1.1
Host: BucketName.s3.amazonaws.com
Date: date
Authorization: authorization string (see Authenticating Requests (AWS Signature Version 4))
Content-MD5: MD5

request body
```

**Note**

The syntax shows some of the request headers. For a complete list, see "Request Headers," later in this topic.

**Request Parameters**

This implementation of the operation does not use request parameters.

**Request Headers**

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
<th>Required</th>
</tr>
</thead>
<tbody>
<tr>
<td>Content-MD5</td>
<td>The base64-encoded 128-bit MD5 digest of the data. You must use this header as a message integrity check to verify that the request body was not corrupted in transit. For more information, see RFC 1864. Type: String</td>
<td>Yes</td>
</tr>
</tbody>
</table>
Amazon Simple Storage Service API Reference
Requests

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
<th>Required</th>
</tr>
</thead>
<tbody>
<tr>
<td>Default: None</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

### Request Elements

The following is an XML example of a request body for restoring an archive.

```xml
<RestoreRequest>
  <Days>2</Days>
  <GlacierJobParameters>
    <Tier>Bulk</Tier>
  </GlacierJobParameters>
</RestoreRequest>
```

The following table explains the XML for archive restoration in the request body.

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
<th>Required</th>
</tr>
</thead>
<tbody>
<tr>
<td>RestoreRequest</td>
<td>Container for restore information.</td>
<td>Yes</td>
</tr>
<tr>
<td></td>
<td>Type: Container</td>
<td></td>
</tr>
</tbody>
</table>
| Days                   | Lifetime of the restored (active) copy. The minimum number of days that you can restore an object from S3 Glacier is 1. After the object copy reaches the specified lifetime, Amazon S3 removes it from the bucket. If you are restoring an archive, this element is required.
|                        | Do not use this element with a SELECT type of request.                      | Yes, if restoring an archive |
|                        | Type: Positive integer                                                      |          |
|                        | Ancestors: RestoreRequest                                                   |          |
| GlacierJobParameters   | Container for Glacier job parameters.                                       | No       |
|                        | Do not use this element with a SELECT type of request.                      |          |
|                        | Type: Container                                                             |          |
|                        | Ancestors: RestoreRequest                                                   |          |
| Tier                   | The data access tier to use when restoring the archive. Standard is the default. | No       |
|                        | Type: Enum                                                                  |          |
|                        | Valid values: Expedited | Standard | Bulk | | |
|                        | Ancestors: GlacierJobParameters                                             |          |

The following XML is the request body for a select query on an archived object:

```xml
<RestoreRequest>
  <Type>SELECT</Type>
</RestoreRequest>
```
<Tier>Expedited</Tier>
<Description>Job description</Description>
<SelectParameters>
    <Expression>Select * from Object</Expression>
    <ExpressionType>SQL</ExpressionType>
    <InputSerialization>
        <CSV>
            <FileHeaderInfo>IGNORE</FileHeaderInfo>
            <RecordDelimiter>\n</RecordDelimiter>
            <FieldDelimiter>,</FieldDelimiter>
            <QuoteCharacter>"</QuoteCharacter>
            <QuoteEscapeCharacter>"</QuoteEscapeCharacter>
            <Comments>#</Comments>
        </CSV>
    </InputSerialization>
    <OutputSerialization>
        <CSV>
            <QuoteFields>ASNEEDED</QuoteFields>
            <RecordDelimiter>\n</RecordDelimiter>
            <FieldDelimiter>,</FieldDelimiter>
            <QuoteCharacter>"</QuoteCharacter>
            <QuoteEscapeCharacter>"</QuoteEscapeCharacter>
        </CSV>
    </OutputSerialization>
</SelectParameters>
<OutputLocation>
    <S3>
        <BucketName>Bucket name</BucketName>
        <Prefix>Key prefix</Prefix>
        <CannedACL>Canned ACL string</CannedACL>
        <AccessControlList>
            <Grantee>
                <Type>Grantee Type</Type>
                <ID>Grantee identifier</ID>
                <URI>Grantee URI</URI>
                <Permission>Granted permission</Permission>
                    <DisplayName>Display Name</DisplayName>
                    <EmailAddress>email</EmailAddress>
            </Grantee>
        </AccessControlList>
        <Encryption>
            <EncryptionType>Encryption type</EncryptionType>
            <KMSKeyId>KMS Key ID</KMSKeyId>
            <KMSContext>Base64-encoded JSON</KMSContext>
        </Encryption>
        <UserMetadata>
            <MetadataEntry>
                <Name>Key</Name>
                <Value>Value</Value>
            </MetadataEntry>
        </UserMetadata>
        <Tagging>
            <TagSet>
                <Tag>
                    <Key>Tag name</Key>
                    <Value>Tag value</Value>
                </Tag>
            </TagSet>
        </Tagging>
    </S3>
</OutputLocation>
</RestoreRequest>

The following tables explain the XML for a SELECT type of restoration in the request body.
### Requests

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
<th>Required</th>
</tr>
</thead>
<tbody>
<tr>
<td>RestoreRequest</td>
<td>Container for restore information.</td>
<td>Yes</td>
</tr>
<tr>
<td>Tier</td>
<td>The data access tier to use when restoring the archive. Standard is the default.</td>
<td>No</td>
</tr>
<tr>
<td>Description</td>
<td>The optional description for the request.</td>
<td>No</td>
</tr>
<tr>
<td>SelectParameters</td>
<td>Describes the parameters for the select job request.</td>
<td>Yes, if request type is SELECT</td>
</tr>
<tr>
<td>OutputLocation</td>
<td>Describes the location that receives the results of the select restore request.</td>
<td>Yes, if request type is SELECT</td>
</tr>
</tbody>
</table>

**The SelectParameters container element contains the following elements.**

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
<th>Required</th>
</tr>
</thead>
<tbody>
<tr>
<td>Expression</td>
<td>The SQL expression. For example:</td>
<td>Yes</td>
</tr>
<tr>
<td></td>
<td>- The following SQL expression retrieves the first column of the data from the object stored in CSV format:</td>
<td></td>
</tr>
<tr>
<td></td>
<td>SELECT s._1 FROM Object s</td>
<td></td>
</tr>
<tr>
<td></td>
<td>- The following SQL expression returns everything from the object:</td>
<td></td>
</tr>
<tr>
<td></td>
<td>SELECT * FROM Object</td>
<td></td>
</tr>
<tr>
<td>ExpressionType</td>
<td>Identifies the expression type.</td>
<td>Yes</td>
</tr>
<tr>
<td></td>
<td>Type: String</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Valid values: SQL</td>
<td></td>
</tr>
</tbody>
</table>
### InputSerialization

Describes the serialization format of the object.

- **Type:** Container for CSV
- **Ancestors:** SelectParameters

### OutputSerialization

Describes how the results of the select job are serialized.

- **Type:** Container for CSV
- **Ancestors:** SelectParameters

---

### The CSV container element in the InputSerialization element contains the following elements.

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
<th>Required</th>
</tr>
</thead>
<tbody>
<tr>
<td>RecordDelimiter</td>
<td>A single character used to separate individual records in the input. Instead of the default value, you can specify an arbitrary delimiter.</td>
<td>No</td>
</tr>
<tr>
<td></td>
<td><strong>Type:</strong> String</td>
<td></td>
</tr>
<tr>
<td></td>
<td><strong>Default:</strong> \n</td>
<td></td>
</tr>
<tr>
<td></td>
<td><strong>Ancestors:</strong> CSV</td>
<td></td>
</tr>
<tr>
<td>FieldDelimiter</td>
<td>A single character used to separate individual fields in a record. You can specify an arbitrary delimiter.</td>
<td>No</td>
</tr>
<tr>
<td></td>
<td><strong>Type:</strong> String</td>
<td></td>
</tr>
<tr>
<td></td>
<td><strong>Default:</strong> ,</td>
<td></td>
</tr>
<tr>
<td></td>
<td><strong>Ancestors:</strong> CSV</td>
<td></td>
</tr>
<tr>
<td>QuoteCharacter</td>
<td>A single character used for escaping when the field delimiter is part of the value. Consider this example in a CSV file: &quot;a, b&quot; Wrapping the value in quotation marks makes this value a single field. If you don't use the quotation marks, the comma is a field delimiter (which makes it two separate field values, a and b).</td>
<td>No</td>
</tr>
<tr>
<td></td>
<td><strong>Type:</strong> String</td>
<td></td>
</tr>
<tr>
<td></td>
<td><strong>Default:</strong> &quot;</td>
<td></td>
</tr>
<tr>
<td></td>
<td><strong>Ancestors:</strong> CSV</td>
<td></td>
</tr>
<tr>
<td>QuoteEscapeCharacter</td>
<td>A single character used for escaping the quotation mark character inside an already escaped value. For example, the value &quot;&quot;&quot;&quot; a , b &quot;&quot;&quot;&quot; is parsed as &quot; a , b &quot;.</td>
<td>No</td>
</tr>
</tbody>
</table>
### FileHeaderInfo

Describes the first line in the input data. It is one of the ENUM values.

- **NONE**: First line is not a header.
- **IGNORE**: First line is a header, but you can't use the header values to indicate the column in an expression. You can use column position (such as `_1, _2, ...`) to indicate the column (SELECT s._1 FROM OBJECT s).
- **USE**: First line is a header, and you can use the header value to identify a column in an expression (SELECT "name" FROM OBJECT).

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
<th>Required</th>
</tr>
</thead>
<tbody>
<tr>
<td>FileHeaderInfo</td>
<td>Describes the first line in the input data. It is one of the ENUM values.</td>
<td>No</td>
</tr>
</tbody>
</table>

Type: Enum

Valid values: **NONE** | **USE** | **IGNORE**

### Comments

A single character used to indicate that a row should be ignored when the character is present at the start of that row. You can specify any character to indicate a comment line.

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
<th>Required</th>
</tr>
</thead>
<tbody>
<tr>
<td>Comments</td>
<td>A single character used to indicate that a row should be ignored when the character is present at the start of that row. You can specify any character to indicate a comment line.</td>
<td>No</td>
</tr>
</tbody>
</table>

Type: String

Ancestors: CSV

---

**The CSV container element (in the OutputSerialization elements) contains the following elements.**

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
<th>Required</th>
</tr>
</thead>
<tbody>
<tr>
<td>QuoteFields</td>
<td>Indicates whether to use quotation marks around output fields.</td>
<td>No</td>
</tr>
</tbody>
</table>

- **ALWAYS**: Always use quotation marks for output fields.
- **ASNEEDED**: Use quotation marks for output fields when needed.

Type: Enum

Valid values: **ALWAYS** | **ASNEEDED**

Default: **ASNEEDED**

Ancestors: CSV

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
<th>Required</th>
</tr>
</thead>
<tbody>
<tr>
<td>RecordDelimiter</td>
<td>A single character used to separate individual records in the output. Instead of the default value, you can specify an arbitrary delimiter.</td>
<td>No</td>
</tr>
<tr>
<td>Name</td>
<td>Description</td>
<td>Required</td>
</tr>
<tr>
<td>-----------------------</td>
<td>------------------------------------------------------------------------------</td>
<td>----------</td>
</tr>
<tr>
<td>FieldDelimiter</td>
<td>A single character used to separate individual fields in a record. You can specify an arbitrary delimiter. Type: String Default: , Ancestors: CSV</td>
<td>No</td>
</tr>
<tr>
<td>QuoteCharacter</td>
<td>A single character used for escaping when the field delimiter is part of the value. For example, if the value is a, b, Amazon S3 wraps this field value in quotation marks, as follows: &quot; a , b &quot;. Type: String Default: &quot; Ancestors: CSV</td>
<td>No</td>
</tr>
<tr>
<td>QuoteEscapeCharacter</td>
<td>A single character used for escaping the quotation mark character inside an already escaped value. For example, if the value is &quot; a , b &quot;, Amazon S3 wraps the value in quotation marks, as follows: &quot;&quot; a , b &quot;&quot;. Type: String Ancestors: CSV</td>
<td>No</td>
</tr>
</tbody>
</table>

The S3 container element (in the OutputLocation element) contains the following elements.

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
<th>Required</th>
</tr>
</thead>
<tbody>
<tr>
<td>AccessControlList</td>
<td>A list of grants that control access to the staged results. Type: Container for Grant Ancestors: S3</td>
<td>No</td>
</tr>
<tr>
<td>BucketName</td>
<td>The name of the S3 bucket where the select restore results are stored. The bucket must be in the same AWS Region as the bucket that contains the input archive object. Type: String Ancestors: S3</td>
<td>Yes</td>
</tr>
<tr>
<td>CannedACL</td>
<td>The canned access control list (ACL) to apply to the select restore results. Type: String Ancestors: CSV</td>
<td>No</td>
</tr>
</tbody>
</table>
### Name | Description | Required
---|---|---
**Encryption** | Contains encryption information for the stored results. Type: Container for Encryption Ancestors: S3 | No

**Prefix** | The prefix that is prepended to the select restore results. The maximum length for the prefix is 512 bytes. Type: String Ancestors: S3 | Yes

**StorageClass** | The class of storage used to store the select request results. Type: String Valid values: STANDARD | REDUCED_REDUNDANCY | STANDARD_IA | ONEZONE_IA Ancestors: S3 | No

**Tagging** | Container for tag information. Type: Tag structure Ancestors: S3 | No

**UserMetadata** | Contains a list of metadata to store with the select restore results. Type: MetadataEntry structure Ancestors: S3 | No

**The Grantee container element (in the AccessControlList element) contains the following elements.**

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
<th>Required</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>DisplayName</strong></td>
<td>The screen name of the grantee. Type: String Ancestors: Grantee</td>
<td>No</td>
</tr>
<tr>
<td><strong>EmailAddress</strong></td>
<td>The email address of the grantee. Type: String Ancestors: Grantee</td>
<td>No</td>
</tr>
<tr>
<td><strong>ID</strong></td>
<td>The canonical user ID of the grantee.</td>
<td>No</td>
</tr>
<tr>
<td>Name</td>
<td>Description</td>
<td>Required</td>
</tr>
<tr>
<td>----------</td>
<td>-------------</td>
<td>----------</td>
</tr>
<tr>
<td>Type</td>
<td>The type of the grantee.</td>
<td>No</td>
</tr>
<tr>
<td></td>
<td>Type: String</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Ancestors: Grantee</td>
<td></td>
</tr>
<tr>
<td>URI</td>
<td>The URI of the grantee group.</td>
<td>No</td>
</tr>
<tr>
<td></td>
<td>Type: String</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Ancestors: Grantee</td>
<td></td>
</tr>
<tr>
<td>Permission</td>
<td>Granted permission.</td>
<td>No</td>
</tr>
<tr>
<td></td>
<td>Type: String</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Ancestors: Grantee</td>
<td></td>
</tr>
</tbody>
</table>

The Encryption container element (in S3) contains the following elements.

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
<th>Required</th>
</tr>
</thead>
<tbody>
<tr>
<td>EncryptionType</td>
<td>The server-side encryption algorithm used when storing job results. The default is no encryption.</td>
<td>No</td>
</tr>
<tr>
<td></td>
<td>Type: String</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Valid Values aws:kms</td>
<td>AES256</td>
</tr>
<tr>
<td></td>
<td>Ancestors: Encryption</td>
<td></td>
</tr>
<tr>
<td>KMSContext</td>
<td>Optional. If the encryption type is aws:kms, you can use this value to specify the encryption context for the select restore results.</td>
<td>No</td>
</tr>
<tr>
<td></td>
<td>Type: String</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Ancestors: Encryption</td>
<td></td>
</tr>
<tr>
<td>KMSKeyId</td>
<td>The AWS Key Management Service (AWS KMS) key ID to use for object encryption.</td>
<td>No</td>
</tr>
<tr>
<td></td>
<td>Type: String</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Ancestors: Encryption</td>
<td></td>
</tr>
</tbody>
</table>

The TagSet container element (in the Tagging element) contains the following element.

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
<th>Required</th>
</tr>
</thead>
<tbody>
<tr>
<td>Tag</td>
<td>Contains tags.</td>
<td>No</td>
</tr>
<tr>
<td></td>
<td>Type: Container</td>
<td></td>
</tr>
</tbody>
</table>
### The Tag container element (in the TagSet element) contains the following elements.

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
<th>Required</th>
</tr>
</thead>
<tbody>
<tr>
<td>Key</td>
<td>Name of the tag.</td>
<td>No</td>
</tr>
<tr>
<td></td>
<td>Type: String</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Ancestors: Tag</td>
<td></td>
</tr>
<tr>
<td>Value</td>
<td>Value of the tag.</td>
<td>No</td>
</tr>
<tr>
<td></td>
<td>Type: String</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Ancestors: Tag</td>
<td></td>
</tr>
</tbody>
</table>

### The MetadataEntry container element (in the UserMetadata element) contains the following key-value pair elements to store with an object.

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
<th>Required</th>
</tr>
</thead>
<tbody>
<tr>
<td>MetadataKey</td>
<td>The metadata key.</td>
<td>No</td>
</tr>
<tr>
<td></td>
<td>Type: String</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Ancestors:</td>
<td></td>
</tr>
<tr>
<td>MetadataEntry</td>
<td>The metadata value.</td>
<td>No</td>
</tr>
<tr>
<td></td>
<td>Type: String</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Ancestors:</td>
<td></td>
</tr>
</tbody>
</table>

### Responses

A successful operation returns either the **200 OK** or **202 Accepted** status code.

- If the object copy is not previously restored, then Amazon S3 returns **202 Accepted** in the response.
- If the object copy is previously restored, Amazon S3 returns **200 OK** in the response.

### Response Headers

This implementation of the operation uses only response headers that are common to most responses. For more information, see Common Response Headers (p. 814).

### Response Elements

This operation does not return response elements.
### Special Errors

<table>
<thead>
<tr>
<th>Error Code</th>
<th>Description</th>
<th>HTTP Status Code</th>
<th>SOAP Fault Code Prefix</th>
</tr>
</thead>
<tbody>
<tr>
<td>RestoreAlreadyInProgress</td>
<td>Object restore is already in progress. (This error does not apply to SELECT type requests.)</td>
<td>409 Conflict</td>
<td>Client</td>
</tr>
<tr>
<td>GlacierExpeditedRetrievalNotAvailable</td>
<td>Glacier expedited retrievals are currently not available. Try again later. (Returned if there is insufficient capacity to process the Expedited request. This error applies only to Expedited retrievals and not to Standard or Bulk retrievals.)</td>
<td>503</td>
<td>N/A</td>
</tr>
</tbody>
</table>

### Examples

**Restore an Object for Two Days Using the Expedited Retrieval Option**

The following restore request restores a copy of the photo1.jpg object from S3 Glacier for a period of two days using the expedited retrieval option.

```
POST /photo1.jpg?restore HTTP/1.1
Host: examplebucket.s3.amazonaws.com
Date: Mon, 22 Oct 2012 01:49:52 GMT
Authorization: authorization string
Content-Length: content length

<RestoreRequest>
  <Days>2</Days>
  <GlacierJobParameters>
    <Tier>Expedited</Tier>
  </GlacierJobParameters>
</RestoreRequest>
```

If the examplebucket does not have a restored copy of the object, Amazon S3 returns the following 202 Accepted response.

```
HTTP/1.1 202 Accepted
x-amz-id-2: GFihv3y6+kE7KG11GEkQhU7/2/chR3Yb2fCb2S04nxI423Dqgw2XiqO8/UElzYqP1BlZNRcovw=
x-amz-request-id: 9F341CD3C4BA79E0
Date: Sat, 20 Oct 2012 23:54:05 GMT
Content-Length: 0
Server: AmazonS3
```

If a copy of the object is already restored, Amazon S3 returns a 200 OK response, and updates only the restored copy's expiry time.

**Query an Archive with a SELECT Request**

The following is an example select restore request.
POST /object-one.csv?restore HTTP/1.1
Host: examplebucket.s3.amazonaws.com
Date: Date: Sat, 20 Oct 2012 23:54:05 GMT
Authorization: authorization string
Content-Length: content length

<RestoreRequest xmlns="http://s3.amazonaws.com/doc/2006-03-01/">
  <Type>SELECT</Type>
  <Tier>Expedited</Tier>
  <Description>this is a description</Description>
  <SelectParameters>
    <InputSerialization>
      <CSV>
        <FileHeaderInfo>IGNORE</FileHeaderInfo>
        <Comments>IgnoreCase</Comments>
        <RecordDelimiter/>
        <FieldDelimiter>,</FieldDelimiter>
        <QuoteCharacter>"</QuoteCharacter>
      </CSV>
    </InputSerialization>
    <ExpressionType>SQL</ExpressionType>
    <Expression>select * from object</Expression>
    <OutputSerialization>
      <CSV>
        <QuoteFields>ALWAYS</QuoteFields>
        <RecordDelimiter/>
        <FieldDelimiter>,</FieldDelimiter>
        <QuoteCharacter>"</QuoteCharacter>
      </CSV>
    </OutputSerialization>
  </SelectParameters>
  <OutputLocation>
    <S3>
      <BucketName>example-output-bucket</BucketName>
      <Prefix>test-s3</Prefix>
      <AccessControlList>
        <Grant>
          <Grantee xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:type="AmazonCustomerByEmail">
            <EmailAddress>jane-doe@example.com</EmailAddress>
          </Grantee>
          <Permission>FULL_CONTROL</Permission>
        </Grant>
        <UserMetadata>
          <MetadataEntry>
            <Name>test</Name>
            <Value>test-value</Value>
          </MetadataEntry>
          <MetadataEntry>
            <Name>other</Name>
            <Value>something else</Value>
          </MetadataEntry>
        </UserMetadata>
        <StorageClass>STANDARD</StorageClass>
      </S3>
    </OutputLocation>
  </RestoreRequest>

Amazon S3 returns the following 202 Accepted response.

HTTP/1.1 202 Accepted
Browser-Based Uploads Using HTTP POST

Amazon S3 supports HTTP POST requests so that users can upload content directly to Amazon S3. By using POST, end users can authenticate requests without having to pass data through a secure intermediary node that protects your credentials. Thus, HTTP POST has the potential to reduce latency.

The following figure shows an Amazon S3 upload using a POST request.

### Uploading Using POST

<p>| | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>The user accesses your page from a web browser.</td>
</tr>
<tr>
<td>2</td>
<td>Your webpage contains an HTML form that contains all the information necessary for the user to upload content to Amazon S3.</td>
</tr>
</tbody>
</table>
The user uploads content to Amazon S3 through the web browser.

The process for sending browser-based POST requests is as follows:

1. Create a security policy specifying conditions that restrict what you want to allow in the request, such as the bucket name where objects can be uploaded, and key name prefixes that you want to allow for the object that is being created.
2. Create a signature that is based on the policy. For authenticated requests, the form must include a valid signature and the policy.
3. Create an HTML form that your users can access in order to upload objects to your Amazon S3 bucket.

The following section describes how to create a signature to authenticate a request. For information about creating forms and security policies, see Creating an HTML Form (Using AWS Signature Version 4) (p. 797).

Calculating a Signature

For authenticated requests, the HTML form must include fields for a security policy and a signature.

- A security policy (see Creating a POST Policy (p. 801)) controls what is allowed in the request.
- The security policy is the StringToSign (see Introduction to Signing Requests (p. 734)) in your signature calculation.

To Calculate a signature

1. Create a policy using UTF-8 encoding.
2. Convert the UTF-8-encoded policy bytes to base64. The result is the StringToSign.
3. Create a signing key.
4. Use the signing key to sign the StringToSign using HMAC-SHA256 signing algorithm.

For more information about creating HTML forms, security policies, and an example, see the following:

- Creating an HTML Form (Using AWS Signature Version 4) (p. 797)
Creating an HTML Form (Using AWS Signature Version 4)

Topics

- HTML Form Declaration (p. 797)
- HTML Form Fields (p. 798)

To allow users to upload content to Amazon S3 by using their browsers (HTTP POST requests), you use HTML forms. HTML forms consist of a form declaration and form fields. The form declaration contains high-level information about the request. The form fields contain detailed request information.

This section describes how to create HTML forms. For a working example of browser-based upload using HTTP POST and related signature calculations for request authentication, see Example: Browser-Based Upload using HTTP POST (Using AWS Signature Version 4) (p. 807).

The form and policy must be UTF-8 encoded. You can apply UTF-8 encoding to the form by specifying charset=UTF-8 in the content attribute. The following is an example of UTF-8 encoding in the HTML heading.

```html
<html>
<head>
  ...
  <meta http-equiv="Content-Type" content="text/html; charset=UTF-8" />
  ...
</head>
<body>

Following is an example of UTF-8 encoding in a request header.

```

Note
The form data and boundaries (excluding the contents of the file) cannot exceed 20KB.

HTML Form Declaration

The HTML form declaration has the following three attributes:

- action – The URL that processes the request, which must be set to the URL of the bucket. For example, if the name of your bucket is examplebucket, the URL is http://examplebucket.s3.amazonaws.com/.

  Note
  The key name is specified in a form field.

- method – The method must be POST.

- enctype – The enclosure type (enctype) must be set to multipart/form-data for both file uploads and text area uploads. For more information about enctype, see RFC 1867.
This is a form declaration for the bucket examplebucket.

```html
<form action="http://examplebucket.s3.amazonaws.com/" method="post"
enctype="multipart/form-data">

HTML Form Fields

The following table describes a list of fields that you can use within a form. Among other fields, there is a signature field that you can use to authenticate requests. There are fields for you to specify the signature calculation algorithm (x-amz-algorithm), the credential scope (x-amz-credential) that you used to generate the signing key, and the date (x-amz-date) used to calculate the signature. Amazon S3 uses this information to re-create the signature. If the signatures match, Amazon S3 processes the request.

**Note**
The variable `${filename}` is automatically replaced with the name of the file provided by the user and is recognized by all form fields. If the browser or client provides a full or partial path to the file, only the text following the last slash (/) or backslash (\) is used (for example, C:\Program Files\directory1\file.txt is interpreted as file.txt). If no file or file name is provided, the variable is replaced with an empty string.

If you don’t provide elements required for authenticated requests, such as the policy element, the request is assumed to be anonymous and will succeed only if you have configured the bucket for public read and write.

<table>
<thead>
<tr>
<th>Element Name</th>
<th>Description</th>
<th>Required</th>
</tr>
</thead>
<tbody>
<tr>
<td>acl</td>
<td>An Amazon S3 access control list (ACL). If an invalid ACL is specified, Amazon S3 denies the request. For more information about ACLs, see <a href="#">Using Amazon S3 ACLs</a>.</td>
<td>No</td>
</tr>
<tr>
<td></td>
<td>Type: String</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Default: private</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Valid Values: private</td>
<td>public-read</td>
</tr>
<tr>
<td>Cache-Control</td>
<td>REST-specific headers. For more information, see <a href="#">PutObject</a> (p. 343).</td>
<td>No</td>
</tr>
<tr>
<td>Content-Type</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Content-Disposition</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Content-Encoding</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Expires</td>
<td></td>
<td></td>
</tr>
<tr>
<td>key</td>
<td>The key name of the uploaded object. To use the file name provided by the user, use the <code>${filename}</code> variable. For example, if you upload a file photo1.jpg and you specify /user/user1/${filename} as key name, the file is stored as /user/user1/photo1.jpg.</td>
<td>Yes</td>
</tr>
<tr>
<td>Element Name</td>
<td>Description</td>
<td>Required</td>
</tr>
<tr>
<td>----------------------</td>
<td>---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
<td>---------------------------------------</td>
</tr>
<tr>
<td>policy</td>
<td>The base64-encoded security policy that describes what is permitted in the request. For authenticated requests, a policy is required._requests without a security policy are considered anonymous and will succeed only on a publicly writable bucket.</td>
<td>Required for authenticated requests</td>
</tr>
</tbody>
</table>
| success_action_redirect | The URL to which the client is redirected upon successful upload.  
If success_action_redirect is not specified, or Amazon S3 cannot interpret the URL, Amazon S3 returns the empty document type that is specified in the success_action_status field.  
If the upload fails, Amazon S3 returns an error and does not redirect the user to another URL.                                                                                                                  | No                                    |
| success_action_status | The status code returned to the client upon successful upload if success_action_redirect is not specified.  
Valid values are 200, 201, or 204 (default).  
If the value is set to 200 or 204, Amazon S3 returns an empty document with the specified status code.  
If the value is set to 201, Amazon S3 returns an XML document with a 201 status code. For information about the content of the XML document, see POST Object (p. 769).  
If the value is not set or is invalid, Amazon S3 returns an empty document with a 204 status code.  
**Note**  
Some versions of the Adobe Flash player do not properly handle HTTP responses with an empty body. To support uploads through Adobe Flash, we recommend setting success_action_status to 201. | No                                    |
| x-amz-algorithm      | The signing algorithm used to authenticate the request. For AWS Signature Version 4, the value is AWS4-HMAC-SHA256.  
This field is required if a policy document is included with the request.                                                                                                                                                                                                 | Required for authenticated requests   |
<table>
<thead>
<tr>
<th>Element Name</th>
<th>Description</th>
<th>Required</th>
</tr>
</thead>
<tbody>
<tr>
<td>x-amz-credential</td>
<td>In addition to your access key ID, this field also provides scope information identifying region and service for which the signature is valid. This should be the same scope you used in calculating the signing key for signature calculation. It is a string of the following form: <code>&lt;your-access-key-id&gt;/&lt;date&gt;/&lt;aws-region&gt;/&lt;aws-service&gt;/aws4_request</code> For example: AKIAIOSFODNN7EXAMPLE/20130728/us-east-1/s3/aws4_request For Amazon S3, the <code>aws-service</code> string is <code>s3</code>. For a list of Amazon S3 <code>aws-region</code> strings, see Regions and Endpoints in the AWS General Reference. This is required if a policy document is included with the request.</td>
<td>Required for authenticated requests</td>
</tr>
<tr>
<td>x-amz-date</td>
<td>It is the date value in ISO8601 format. For example, <code>20130728T000000Z</code>. It is the same date you used in creating the signing key (for example, 20130728). This must also be the same value you provide in the policy <code>x-amz-date</code> that you signed. This is required if a policy document is included with the request.</td>
<td>Required for authenticated requests</td>
</tr>
<tr>
<td>x-amz-security-token</td>
<td>A security token used by Amazon DevPay and session credentials If the request is using Amazon DevPay, it requires two <code>x-amz-security-token</code> form fields: one for the product token and one for the user token. For more information, see Using DevPay in the Amazon Simple Storage Service Developer Guide. If the request is using session credentials, it requires one <code>x-amz-security-token</code> form. For more information, see Requesting Temporary Security Credentials in the IAM User Guide.</td>
<td>No</td>
</tr>
<tr>
<td>x-amz-signature</td>
<td>(AWS Signature Version 4) The HMAC-SHA256 hash of the security policy. This field is required if a policy document is included with the request.</td>
<td>Required for authenticated requests</td>
</tr>
<tr>
<td>Element Name</td>
<td>Description</td>
<td>Required</td>
</tr>
<tr>
<td>--------------</td>
<td>-------------</td>
<td>----------</td>
</tr>
<tr>
<td>x-amz-meta-*</td>
<td>Field names starting with this prefix are user-defined metadata. Each one is stored and returned as a set of key-value pairs. Amazon S3 doesn't validate or interpret user-defined metadata. For more information, see PutObject (p. 343).</td>
<td>No</td>
</tr>
<tr>
<td>x-amz-*</td>
<td>See POST Object (POST Object (p. 769) for other x-amz-* headers.</td>
<td>No</td>
</tr>
<tr>
<td>file</td>
<td>File or text content. The file or content must be the last field in the form. You cannot upload more than one file at a time.</td>
<td>Yes</td>
</tr>
</tbody>
</table>

Conditional items are required for authenticated requests and are optional for anonymous requests.

Now that you know how to create forms, next you can create a security policy that you can sign. For more information, see Creating a POST Policy (p. 801).

## Creating a POST Policy

### Topics

- Expiration (p. 802)
- Condition Matching (p. 802)
- Conditions (p. 803)
- Character Escaping (p. 805)

The policy required for making authenticated requests using HTTP POST is a UTF-8 and base64-encoded document written in JavaScript Object Notation (JSON) that specifies conditions that the request must meet. Depending on how you design your policy document, you can control the access granularity per-upload, per-user, for all uploads, or according to other designs that meet your needs.

This section describes the POST policy. For example signature calculations using POST policy, see Example: Browser-Based Upload using HTTP POST (Using AWS Signature Version 4) (p. 807).

**Note**

Although the policy document is optional, we highly recommend that you use one in order to control what is allowed in the request. If you make the bucket publicly writable, you have no control at all over which users can write to your bucket.

The following is an example of a POST policy document.

```json
{ "expiration": "2007-12-01T12:00:00.000Z",  
"conditions": [  
{"acl": "public-read" },  
{"bucket": "johnsmith" },  
["starts-with", "$key", "user/eric/"]  
] }
```
The POST policy always contains the expiration and conditions elements. The example policy uses two condition matching types (exact matching and starts-with matching). The following sections describe these elements.

### Expiration

The expiration element specifies the expiration date and time of the POST policy in ISO8601 GMT date format. For example, `2013-08-01T12:00:00.000Z` specifies that the POST policy is not valid after midnight GMT on August 1, 2013.

### Condition Matching

Following is a table that describes condition matching types that you can use to specify POST policy conditions (described in the next section). Although you must specify one condition for each form field that you specify in the form, you can create more complex matching criteria by specifying multiple conditions for a form field.

<table>
<thead>
<tr>
<th>Condition Match Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Exact Matches</td>
<td>The form field value must match the value specified. This example indicates that the ACL must be set to public-read:</td>
</tr>
<tr>
<td></td>
<td>{&quot;acl&quot;: &quot;public-read&quot; }</td>
</tr>
<tr>
<td></td>
<td>This example is an alternate way to indicate that the ACL must be set to public-read:</td>
</tr>
<tr>
<td></td>
<td>[ &quot;eq&quot;, &quot;$acl&quot;, &quot;public-read&quot; ]</td>
</tr>
<tr>
<td>Starts With</td>
<td>The value must start with the specified value. This example indicates that the object key must start with user/user1:</td>
</tr>
<tr>
<td></td>
<td>[&quot;starts-with&quot;, &quot;$key&quot;, &quot;user/user1/&quot;]</td>
</tr>
<tr>
<td>Matching Content-Types in a Comma-Separated List</td>
<td>Content-Types values for a starts-with condition that include commas are interpreted as lists. Each value in the list must meet the condition for the whole condition to pass. For example, given the following condition:</td>
</tr>
<tr>
<td></td>
<td>[&quot;starts-with&quot;, &quot;$Content-Type&quot;, &quot;image/&quot;]</td>
</tr>
<tr>
<td></td>
<td>The following value would pass the condition:</td>
</tr>
<tr>
<td></td>
<td>&quot;image/jpg,image/png,image/gif&quot;</td>
</tr>
<tr>
<td></td>
<td>The following value would not pass the condition:</td>
</tr>
<tr>
<td></td>
<td>[&quot;image/jpg,text/plain&quot;]</td>
</tr>
</tbody>
</table>

**Note**

Data elements other than Content-Type are treated as strings, regardless of the presence of commas.
The specific conditions supported in a POST policy are described in Conditions (p. 803).

**Conditions**

The conditions in a POST policy is an array of objects, each of which is used to validate the request. You can use these conditions to restrict what is allowed in the request. For example, the preceding policy conditions require the following:

- Request must specify the johnsmith bucket name.
- Object key name must have the user/eric prefix.
- Object ACL must be set to public-read.

Each form field that you specify in a form (except x-amz-signature, file, policy, and field names that have an x-ignore- prefix) must appear in the list of conditions.

**Note**

All variables within the form are expanded prior to validating the POST policy. Therefore, all condition matching should be against the expanded form fields. Suppose that you want to restrict your object key name to a specific prefix (user/user1). In this case, you set the key form field to user/user1/${filename}. Your POST policy should be `[ "starts-with", "$key", "user/user1/" ]` (do not enter `[ "starts-with", "$key", "user/user1/ ${filename}" ]`). For more information, see Condition Matching (p. 802).

Policy document conditions are described in the following table.

<table>
<thead>
<tr>
<th>Element Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>acl</td>
<td>Specifies the ACL value that must be used in the form submission.</td>
</tr>
<tr>
<td></td>
<td>This condition supports exact matching and starts-with condition match type discussed in the following section.</td>
</tr>
<tr>
<td>bucket</td>
<td>Specifies the acceptable bucket name.</td>
</tr>
<tr>
<td></td>
<td>This condition supports exact matching condition match type.</td>
</tr>
<tr>
<td>content-length-range</td>
<td>The minimum and maximum allowable size for the uploaded content.</td>
</tr>
<tr>
<td>Element Name</td>
<td>Description</td>
</tr>
<tr>
<td>------------------------------</td>
<td>---------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td><strong>Condition</strong></td>
<td>This condition supports <strong>content-length-range</strong> condition match type.</td>
</tr>
<tr>
<td><strong>Cache-Control</strong></td>
<td>REST-specific headers. For more information, see POST Object (p. 769).</td>
</tr>
<tr>
<td><strong>Content-Type</strong></td>
<td>This condition supports exact matching and <strong>starts-with</strong> condition match type.</td>
</tr>
<tr>
<td><strong>Content-Disposition</strong></td>
<td></td>
</tr>
<tr>
<td><strong>Content-Encoding</strong></td>
<td></td>
</tr>
<tr>
<td><strong>Expires</strong></td>
<td></td>
</tr>
<tr>
<td><strong>key</strong></td>
<td>The acceptable key name or a prefix of the uploaded object. This condition supports exact matching and <strong>starts-with</strong> condition match type.</td>
</tr>
<tr>
<td><strong>success_action_redirect</strong></td>
<td>The URL to which the client is redirected upon successful upload. This condition supports exact matching and <strong>starts-with</strong> condition match type.</td>
</tr>
<tr>
<td><strong>redirect</strong></td>
<td></td>
</tr>
<tr>
<td><strong>success_action_status</strong></td>
<td>The status code returned to the client upon successful upload if <strong>success_action_redirect</strong> is not specified. This condition supports exact matching.</td>
</tr>
<tr>
<td><strong>x-amz-algorithm</strong></td>
<td>The signing algorithm that must be used during signature calculation. For AWS Signature Version 4, the value is <strong>AWS4-HMAC-SHA256</strong>. This condition supports exact matching.</td>
</tr>
</tbody>
</table>
| **x-amz-credential**         | The credentials that you used to calculate the signature. It provides access key ID and scope information identifying region and service for which the signature is valid. This should be the same scope you used in calculating the signing key for signature calculation. It is a string of the following form:  

```
<your-access-key-id>/<date>/<aws-region>/<aws-service>/aws4_request
```

For example:

```
AKIAIOSFODNN7EXAMPLE/20130728/us-east-1/s3/aws4_request
```

For Amazon S3, the aws-service string is **s3**. For a list of Amazon S3 aws-region strings, see Regions and Endpoints in the AWS General Reference. This is required if a POST policy document is included with the request. This condition supports exact matching. |
### Character Escaping

Characters that must be escaped within a POST policy document are described in the following table.

<table>
<thead>
<tr>
<th>Escape Sequence</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>\</td>
<td>Backslash</td>
</tr>
<tr>
<td>$</td>
<td>Dollar symbol</td>
</tr>
<tr>
<td>\b</td>
<td>Backspace</td>
</tr>
<tr>
<td>\f</td>
<td>Form feed</td>
</tr>
<tr>
<td>\n</td>
<td>New line</td>
</tr>
<tr>
<td>\r</td>
<td>Carriage return</td>
</tr>
</tbody>
</table>

#### Note

If your toolkit adds more form fields (for example, Flash adds `filename`), you must add them to the POST policy document. If you can control this functionality, prefix `x-ignore-` to the field so Amazon S3 ignores the feature and it won't affect future versions of this feature.
<table>
<thead>
<tr>
<th>Escape Sequence</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>\t</td>
<td>Horizontal tab</td>
</tr>
<tr>
<td>\v</td>
<td>Vertical tab</td>
</tr>
<tr>
<td>\uxxxxx</td>
<td>All Unicode characters</td>
</tr>
</tbody>
</table>

Now that you are acquainted with forms and policies, and understand how signing works, you can try a POST upload example. You need to write the code to calculate the signature. The example provides a sample form, and a POST policy that you can use to test your signature calculations. For more information, see Example: Browser-Based Upload using HTTP POST (Using AWS Signature Version 4) (p. 807).
Example: Browser-Based Upload using HTTP POST (Using AWS Signature Version 4)

This section shows an example of using an HTTP POST request to upload content directly to Amazon S3.

For more information on Signature Version 4, see Signature Version 4 Signing Process.

Uploading a File to Amazon S3 Using HTTP POST

This example provides a sample POST policy and a form that you can use to upload a file. The topic uses the example policy and fictitious credentials to show you the workflow and resulting signature and policy hash. You can use this data as test suite to verify your signature calculation code.

The example uses the following example credentials the signature calculations. You can use these credentials to verify your signature calculation code. However, you must then replace these with your own credentials when sending requests to AWS.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>AWSAccessKeyId</td>
<td>AKIAIOSFODNN7EXAMPLE</td>
</tr>
<tr>
<td>AWSSecretAccessKey</td>
<td>wJalrXUtnFEMI/K7MDENG/bPxRfiCYEXAMPLEKEY</td>
</tr>
</tbody>
</table>

Sample Policy and Form

The following POST policy supports uploads to Amazon S3 with specific conditions.

```json
{
    "expiration": "2015-12-30T12:00:00.000Z",
    "conditions": [
        {"bucket": "sigv4examplebucket"},
        {"starts-with": "$key", "user/user1"},
        {"acl": "public-read"},
        {"success_action_redirect": "http://sigv4examplebucket.s3.amazonaws.com/successful_upload.html"},
        {"starts-with": "$Content-Type", "image/"},
        {"x-amz-meta-uuid": "14365123651274"},
        {"x-amz-server-side-encryption": "AES256"},
        {"starts-with": "$x-amz-meta-tag", ""},
        {"x-amz-credential": "AKIAIOSFODNN7EXAMPLE/20151229/us-east-1/s3/aws4_request"},
        {"x-amz-algorithm": "AWS4-HMAC-SHA256"},
        {"x-amz-date": "20151229T000000Z" }
    ]
}
```

This POST policy sets the following conditions on the request:

- The upload must occur before noon UTC on December 30, 2015.
- The content can be uploaded only to the `sigv4examplebucket`. The bucket must be in the region that you specified in the credential scope (`x-amz-credential` form parameter), because the signature you provided is valid only within this scope.
- You can provide any key name that starts with `user/user1`. For example, `user/user1/MyPhoto.jpg`.
• The ACL must be set to public-read.
• If the upload succeeds, the user’s browser is redirected to http://sigv4examplebucket.s3.amazonaws.com/successful_upload.html.
• The object must be an image file.
• The x-amz-meta-uuid tag must be set to 14365123651274.
• The x-amz-meta-tag can contain any value.

The following is a Base64-encoded version of this POST policy. You use this value as your StringToSign in signature calculation.

eyAiZXhwaXJhdGlvbiI6ICIyMDE1LTEyLTMwVDEyOjAwOjAwLjAwMFoiLA0KICAiY29uZGl0aW9ucyI6IFsNCiAgICB7ImJ1Y2tldCI6ICJzaWd2NGV4YW1wbG ... AgIHsieC1hbXotYWxnb3JpdGhtIjogIkFXUzQtSE1BQy1TSEEyNTYifSwNCiAgICB7IngtYW16LWRhdGUiOiAiMjAxNTEyMjlUMDAwMDAwWiIgfQ0KICBdDQp9

When you copy/paste the preceding policy, it should have carriage returns and new lines for your computed hash to match this value (ie. ASCII text, with CRLF line terminators).

Using example credentials to create a signature, the signature value is as follows (in signature calculation, the date is same as the x-amz-date in the policy (20151229):

8afdbf4008c03f22c2c3d3c72e4afbb1f6a588f3255ac628749a66d7f09699e

The following example form specifies the preceding POST policy and supports a POST request to the sigv4examplebucket. Copy/paste the content in a text editor and save it as exampleform.html. You can then upload image files to the specific bucket using the exampleform.html. Your request will succeed if the signature you provide matches the signature Amazon S3 calculates.

Note
You must update the bucket name, dates, credential, policy, and signature with valid values for this to successfully upload to S3.

<html>
<head>
    <meta http-equiv="Content-Type" content="text/html; charset=UTF-8" />
</head>
<body>
<form action="http://sigv4examplebucket.s3.amazonaws.com/" method="post" enctype="multipart/form-data">
    Key to upload:
    <input type="input" name="key" value="user/user1/${filename}" />
    <input type="hidden" name="acl" value="public-read" />
    <input type="hidden" name="success_action_redirect" value="http://sigv4examplebucket.s3.amazonaws.com/successful_upload.html" />
    Content-Type:
    <input type="input" name="Content-Type" value="image/jpeg" />
    <input type="hidden" name="x-amz-meta-uuid" value="14365123651274" />
    <input type="hidden" name="x-amz-server-side-encryption" value="AES256" />
    <input type="text" name="X-Amz-Credential" value="AKIAIOSFODNN7EXAMPLE/20151229/us-east-1/s3/aws4_request" />
    <input type="text" name="X-Amz-Algorithm" value="AWS4-HMAC-SHA256" />
    <input type="text" name="X-Amz-Date" value="20151229T000000Z" />
    Tags for File:
    <input type="input" name="x-amz-meta-tag" value="" />
    <input type="hidden" name="Policy" value="<Base64-encoded policy string>" />
    <input type="hidden" name="X-Amz-Signature" value="<signature-value>" />
    File:
Using POST with Adobe Flash to Upload Objects

This section discusses uploading objects with an HTTP POST request when using Adobe Flash.

Using POST with Adobe Flash

This section describes how to use POST with Adobe Flash.

Adobe Flash Player Security

By default, the Adobe Flash Player security model prohibits making network connections to servers outside the domain that serves the Adobe Flash (.swf) file.

To override the default, you must upload a publicly readable crossdomain.xml file to the bucket that will accept POST uploads. Here is a sample crossdomain.xml file:

```xml
<?xml version="1.0"?>
<!DOCTYPE cross-domain-policy SYSTEM "http://www.macromedia.com/xml/dtds/cross-domain-policy.dtd">
<cross-domain-policy>
  <allow-access-from domain="*" secure="false"/>
</cross-domain-policy>
```

For more information about the Adobe Flash security model, go to the Adobe web site.

When you add the crossdomain.xml file to your bucket, any Adobe Flash Player can connect to the crossdomain.xml file within your bucket. However, crossdomain.xml does not grant access to the Amazon S3 bucket.

Other Adobe Flash Considerations

The FileReference class in the Adobe Flash API adds the Filename form field to the POST request. When you build an Adobe Flash application that uploads files to Amazon S3 by using the FileReference class, include the following condition in your policy:

```javascript
['starts-with', '$Filename', '']
```

Some versions of the Adobe Flash Player do not properly handle HTTP responses that have an empty body. To configure POST to return a response that does not have an empty body, set success_action_status to 201. Then, Amazon S3 will return an XML document with a 201 status code. For information about using this as an optional element (currently the only allowed value is the content of the XML document), see POST Object (p. 769). For information about form fields, see HTML Form Fields (p. 798).
Browser-Based Uploads to Amazon S3 Using the AWS Amplify Library

This section describes how to upload files to Amazon S3 using the AWS Amplify JavaScript library.

For information about setting up the AWS Amplify library, see AWS Amplify Installation and Configuration.

Using the AWS Amplify JavaScript library to Upload Files to Amazon S3

The AWS Amplify library Storage module gives a simple browser-based upload mechanism for managing user content in public or private Amazon S3 storage.

Example: AWS Amplify Manual Setup

The following example shows the manual setup for using the AWS Amplify Storage module. The default implementation of the Storage module uses Amazon S3.

```javascript
import Amplify from 'aws-amplify';
Amplify.configure(
    Auth: {
        identityPoolId: 'XX-XXXX-X:XXXXXXXX-XXXX-1234-abcd-1234567890ab', //REQUIRED - Amazon Cognito Identity Pool ID
        region: 'XX-XXXX-X', // REQUIRED - Amazon Cognito Region
        userPoolId: 'XX-XXXX-X_abcd1234', //OPTIONAL - Amazon Cognito User Pool ID
        userPoolWebClientId: 'XX-XXXX-X_abcd1234', //OPTIONAL - Amazon Cognito Web Client ID
    },
    Storage: {
        bucket: '', //REQUIRED - Amazon S3 bucket
        region: 'XX-XXXX-X', //OPTIONAL - Amazon service region
    }
);
```

Example: Put data into Amazon S3

The following example shows how to put public data into Amazon S3.

```javascript
Storage.put('test.txt', 'Hello')
    .then (result => console.log(result))
    .catch(err => console.log(err));
```

The following example shows how to put private data into Amazon S3.

```javascript
Storage.put('test.txt', 'Private Content', {
    level: 'private',
    contentType: 'text/plain'
})
    .then (result => console.log(result))
    .catch(err => console.log(err));
```

For more information about using the AWS Amplify Storage module, see AWS Amplify Storage.
More Info

AWS Amplify Quick Start
Common Request Headers

The following table describes headers that can be used by various types of Amazon S3 REST requests.

<table>
<thead>
<tr>
<th>Header Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Authorization</td>
<td>The information required for request authentication. For more information, go to The Authentication Header in the Amazon Simple Storage Service Developer Guide. For anonymous requests this header is not required.</td>
</tr>
<tr>
<td>Content-Length</td>
<td>Length of the message (without the headers) according to RFC 2616. This header is required for PUTs and operations that load XML, such as logging and ACLs.</td>
</tr>
<tr>
<td>Content-Type</td>
<td>The content type of the resource in case the request content in the body. Example: text/plain</td>
</tr>
<tr>
<td>Content-MD5</td>
<td>The base64 encoded 128-bit MD5 digest of the message (without the headers) according to RFC 1864. This header can be used as a message integrity check to verify that the data is the same data that was originally sent. Although it is optional, we recommend using the Content-MD5 mechanism as an end-to-end integrity check. For more information about REST request authentication, go to REST Authentication in the Amazon Simple Storage Service Developer Guide.</td>
</tr>
<tr>
<td>Date</td>
<td>The current date and time according to the requester. Example: Wed, 01 Mar 2006 12:00:00 GMT. When you specify the Authorization header, you must specify either the x-amz-date or the Date header.</td>
</tr>
<tr>
<td>Expect</td>
<td>When your application uses 100-continue, it does not send the request body until it receives an acknowledgment. If the message is rejected based on the headers, the body of the message is not sent. This header can be used only if you are sending a body. Valid Values: 100-continue</td>
</tr>
<tr>
<td>Host</td>
<td>For path-style requests, the value is s3.amazonaws.com. For virtual-style requests, the value is BucketName.s3.amazonaws.com. For more information, go to Virtual Hosting in the Amazon Simple Storage Service Developer Guide. This header is required for HTTP 1.1 (most toolkits add this header automatically); optional for HTTP/1.0 requests.</td>
</tr>
<tr>
<td>x-amz-content-sha256</td>
<td>When using signature version 4 to authenticate request, this header provides a hash of the request payload. For more information see Signature Calculations for the Authorization Header: Transferring Payload in a Single Chunk (AWS Signature Version 4) (p. 737). When uploading object in chunks, you set the value to STREAMING-AWS4-HMAC-SHA256-PAYLOAD to indicate that the signature covers only headers and that there is</td>
</tr>
</tbody>
</table>
### Header Name | Description
--- | ---
| **no payload. For more information, see** [Signature Calculations for the Authorization Header: Transferring Payload in Multiple Chunks (Chunked Upload) (AWS Signature Version 4)](p. 748). |  
| **x-amz-date** | The current date and time according to the requester. Example: Wed, 01 Mar 2006 12:00:00 GMT. When you specify the Authorization header, you must specify either the `x-amz-date` or the `Date` header. If you specify both, the value specified for the `x-amz-date` header takes precedence.  
| **x-amz-security-token** | This header can be used in the following scenarios:  
- Provide security tokens for Amazon DevPay operations - Each request that uses Amazon DevPay requires two `x-amz-security-token` headers: one for the product token and one for the user token. When Amazon S3 receives an authenticated request, it compares the computed signature with the provided signature. Improperly formatted multi-value headers used to calculate a signature can cause authentication issues.  
- Provide security token when using temporary security credentials - When making requests using temporary security credentials you obtained from IAM you must provide a security token using this header. To learn more about temporary security credentials, go to [Making Requests](#).  
| | This header is required for requests that use Amazon DevPay and requests that are signed using temporary security credentials. |
## Common Response Headers

The following table describes response headers that are common to most AWS S3 responses.

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Content-Length</td>
<td>The length in bytes of the body in the response.</td>
</tr>
<tr>
<td></td>
<td>Type: String</td>
</tr>
<tr>
<td></td>
<td>Default: None</td>
</tr>
<tr>
<td>Content-Type</td>
<td>The MIME type of the content. For example, Content-Type: text/html; charset=utf-8</td>
</tr>
<tr>
<td></td>
<td>Type: String</td>
</tr>
<tr>
<td></td>
<td>Default: None</td>
</tr>
<tr>
<td>Connection</td>
<td>specifies whether the connection to the server is open or closed.</td>
</tr>
<tr>
<td></td>
<td>Type: Enum</td>
</tr>
<tr>
<td></td>
<td>Valid Values: open</td>
</tr>
<tr>
<td></td>
<td>Default: None</td>
</tr>
<tr>
<td>Date</td>
<td>The date and time Amazon S3 responded, for example, Wed, 01 Mar 2006 12:00:00 GMT.</td>
</tr>
<tr>
<td></td>
<td>Type: String</td>
</tr>
<tr>
<td></td>
<td>Default: None</td>
</tr>
<tr>
<td>ETag</td>
<td>The entity tag is a hash of the object. The ETag reflects changes only to the contents of an object, not its metadata. The ETag may or may not be an MD5 digest of the object data. Whether or not it is depends on how the object was created and how it is encrypted as described below:</td>
</tr>
<tr>
<td></td>
<td>- Objects created by the PUT Object, POST Object, or Copy operation, or through the AWS Management Console, and are encrypted by SSE-S3 or plaintext, have ETags that are an MD5 digest of their object data.</td>
</tr>
<tr>
<td></td>
<td>- Objects created by the PUT Object, POST Object, or Copy operation, or through the AWS Management Console, and are encrypted by SSE-C or SSE-KMS, have ETags that are not an MD5 digest of their object data.</td>
</tr>
<tr>
<td></td>
<td>- If an object is created by either the Multipart Upload or Part Copy operation, the ETag is not an MD5 digest, regardless of the method of encryption.</td>
</tr>
<tr>
<td></td>
<td>Type: String</td>
</tr>
<tr>
<td>Server</td>
<td>The name of the server that created the response.</td>
</tr>
<tr>
<td></td>
<td>Type: String</td>
</tr>
<tr>
<td></td>
<td>Default: AmazonS3</td>
</tr>
<tr>
<td>Name</td>
<td>Description</td>
</tr>
<tr>
<td>------------------</td>
<td>----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
</tbody>
</table>
| x-amz-delete-marker | Specifies whether the object returned was (true) or was not (false) a delete marker.  
Type: Boolean  
Valid Values: true | false  
Default: false |
| x-amz-id-2       | A special token that is used together with the x-amz-request-id header to help AWS troubleshoot problems. For information about AWS support using these request IDs, see Troubleshooting Amazon S3.  
Type: String  
Default: None |
| x-amz-request-id | A value created by Amazon S3 that uniquely identifies the request. This value is used together with the x-amz-id-2 header to help AWS troubleshoot problems. For information about AWS support using these request IDs, see Troubleshooting Amazon S3.  
Type: String  
Default: None |
| x-amz-version-id | The version of the object. When you enable versioning, Amazon S3 generates a random number for objects added to a bucket. The value is UTF-8 encoded and URL ready. When you PUT an object in a bucket where versioning has been suspended, the version ID is always null.  
Type: String  
Valid Values: null | any URL-ready, UTF-8 encoded string  
Default: null |
Error Responses

This section provides reference information about Amazon S3 errors.

**Note**
SOAP support over HTTP is deprecated, but it is still available over HTTPS. New Amazon S3 features will not be supported for SOAP. We recommend that you use either the REST API or the AWS SDKs.

**Topics**
- REST Error Responses (p. 816)
- List of Error Codes (p. 817)
- List of SELECT Object Content Error Codes (p. 825)
- List of Replication-Related Error Codes (p. 832)
- List of Amazon S3 on Outposts Error Codes (p. 833)

**REST Error Responses**

When an error occurs, the header information contains the following:

- **Content-Type:** application/xml
- An appropriate 3xx, 4xx, or 5xx HTTP status code

The body or the response also contains information about the error. The following sample error response shows the structure of response elements common to all REST error responses.

```xml
<?xml version="1.0" encoding="UTF-8"?>
<Error>
  <Code>NoSuchKey</Code>
  <Message>The resource you requested does not exist</Message>
  <Resource>/mybucket/myfoto.jpg</Resource>
  <RequestId>4442587FB7D0A2F9</RequestId>
</Error>
```

The following table explains the REST error response elements.

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Code</strong></td>
<td>The error code is a string that uniquely identifies an error condition. It is meant to be read and understood by programs that detect and handle errors by type. For more information, see List of Error Codes (p. 817).</td>
</tr>
<tr>
<td></td>
<td>Type: String</td>
</tr>
<tr>
<td></td>
<td>Ancestor: Error</td>
</tr>
<tr>
<td><strong>Error</strong></td>
<td>Container for all error elements.</td>
</tr>
<tr>
<td></td>
<td>Type: Container</td>
</tr>
<tr>
<td></td>
<td>Ancestor: None</td>
</tr>
<tr>
<td><strong>Message</strong></td>
<td>The error message contains a generic description of the error condition in English. It is intended for a human audience. Simple programs display the message directly to</td>
</tr>
</tbody>
</table>
Many error responses contain additional structured data meant to be read and understood by a developer diagnosing programming errors. For example, if you send a Content-MD5 header with a REST PUT request that doesn’t match the digest calculated on the server, you receive a BadDigest error. The error response also includes as detail elements the digest we calculated, and the digest you told us to expect. During development, you can use this information to diagnose the error. In production, a well-behaved program might include this information in its error log.

For information about general response elements, go to Error Responses.

### List of Error Codes

The following table lists Amazon S3 error codes.

<table>
<thead>
<tr>
<th>Error Code</th>
<th>Description</th>
<th>HTTP Status Code</th>
<th>SOAP Fault Code Prefix</th>
</tr>
</thead>
<tbody>
<tr>
<td>AccessDenied</td>
<td>Access Denied</td>
<td>403 Forbidden</td>
<td>Client</td>
</tr>
<tr>
<td>AccountProblem</td>
<td>There is a problem with your AWS account that prevents the operation from completing successfully. For further assistance, Contact Us.</td>
<td>403 Forbidden</td>
<td>Client</td>
</tr>
<tr>
<td>AllAccessDisabled</td>
<td>All access to this Amazon S3 resource has been disabled. For further assistance, Contact Us.</td>
<td>403 Forbidden</td>
<td>Client</td>
</tr>
<tr>
<td>AmbiguousGrantByEmailAddress</td>
<td>The email address you provided is associated with more than one account.</td>
<td>400 Bad Request</td>
<td>Client</td>
</tr>
<tr>
<td>Error Code</td>
<td>Description</td>
<td>HTTP Status Code</td>
<td>SOAP Fault Code Prefix</td>
</tr>
<tr>
<td>------------------------------------</td>
<td>-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
<td>------------------</td>
<td>------------------------</td>
</tr>
<tr>
<td>AuthorizationHeaderMalformed</td>
<td>The authorization header you provided is not valid.</td>
<td>400 Bad Request</td>
<td>N/A</td>
</tr>
<tr>
<td>BadDigest</td>
<td>The Content-MD5 you specified did not match what we received.</td>
<td>400 Bad Request</td>
<td>Client</td>
</tr>
<tr>
<td>BucketAlreadyExists</td>
<td>The requested bucket name is not available. The bucket namespace is shared by all users of the system. Select a different name and try again.</td>
<td>409 Conflict</td>
<td>Client</td>
</tr>
<tr>
<td>BucketAlreadyOwnedByYou</td>
<td>The bucket you tried to create already exists, and you own it. Amazon S3 returns this error in all AWS Regions except us-east-1 (N. Virginia). For legacy compatibility, if you re-create an existing bucket that you already own in us-east-1, Amazon S3 returns 200 OK and resets the bucket access control lists (ACLs). For Amazon S3 on Outposts, the bucket you tried to create already exists in your Outpost and you own it.</td>
<td>409 Conflict</td>
<td>Client</td>
</tr>
<tr>
<td>BucketNotEmpty</td>
<td>The bucket you tried to delete is not empty.</td>
<td>409 Conflict</td>
<td>Client</td>
</tr>
<tr>
<td>CredentialsNotSupported</td>
<td>This request does not support credentials.</td>
<td>400 Bad Request</td>
<td>Client</td>
</tr>
<tr>
<td>CrossLocationLoggingProhibited</td>
<td>Cross-location logging not allowed. Buckets in one geographic location cannot log information to a bucket in another location.</td>
<td>403 Forbidden</td>
<td>Client</td>
</tr>
<tr>
<td>EntityTooSmall</td>
<td>Your proposed upload is smaller than the minimum allowed object size.</td>
<td>400 Bad Request</td>
<td>Client</td>
</tr>
<tr>
<td>EntityTooLarge</td>
<td>Your proposed upload exceeds the maximum allowed object size.</td>
<td>400 Bad Request</td>
<td>Client</td>
</tr>
<tr>
<td>ExpiredToken</td>
<td>The provided token has expired.</td>
<td>400 Bad Request</td>
<td>Client</td>
</tr>
<tr>
<td>IllegalLocationConstraintException</td>
<td>Indicates that you are trying to access a bucket from a different Region than where the bucket exists. To avoid this error, use the --region option. For example: aws s3 cp awsexample.txt s3://testbucket/ --region ap-east-1.</td>
<td>400 Bad Request</td>
<td>Client</td>
</tr>
<tr>
<td>Error Code</td>
<td>Description</td>
<td>HTTP Status Code</td>
<td>SOAP Fault Code Prefix</td>
</tr>
<tr>
<td>------------------------------------------------</td>
<td>-----------------------------------------------------------------------------</td>
<td>------------------</td>
<td>------------------------</td>
</tr>
<tr>
<td>IllegalVersioningConfigurationException</td>
<td>Indicates that the versioning configuration specified in the request is invalid.</td>
<td>400 Bad Request</td>
<td>Client</td>
</tr>
<tr>
<td>IncompleteBody</td>
<td>You did not provide the number of bytes specified by the Content-Length HTTP header.</td>
<td>400 Bad Request</td>
<td>Client</td>
</tr>
<tr>
<td>IncorrectNumberOfFilesInPostRequest</td>
<td>POST requires exactly one file upload per request.</td>
<td>400 Bad Request</td>
<td>Client</td>
</tr>
<tr>
<td>InlineDataTooLarge</td>
<td>Inline data exceeds the maximum allowed size.</td>
<td>400 Bad Request</td>
<td>Client</td>
</tr>
<tr>
<td>InternalError</td>
<td>We encountered an internal error. Please try again.</td>
<td>500 Internal Server Error</td>
<td>Server</td>
</tr>
<tr>
<td>InvalidAccessKeyId</td>
<td>The AWS access key ID you provided does not exist in our records.</td>
<td>403 Forbidden</td>
<td>Client</td>
</tr>
<tr>
<td>InvalidAccessPoint</td>
<td>The specified access point name or account is not valid.</td>
<td>400 Bad Request</td>
<td>Client</td>
</tr>
<tr>
<td>InvalidAddressingHeader</td>
<td>You must specify the Anonymous role.</td>
<td>N/A</td>
<td>Client</td>
</tr>
<tr>
<td>InvalidArgument</td>
<td>This error might occur for the following reasons:</td>
<td>400 Bad Request</td>
<td>Client</td>
</tr>
<tr>
<td></td>
<td>• The specified argument was invalid.</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>• The request was missing a required header.</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>• The specified argument was incomplete or in the wrong format.</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>• Must have length greater than or equal to 3.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>InvalidBucketName</td>
<td>The specified bucket is not valid.</td>
<td>400 Bad Request</td>
<td>Client</td>
</tr>
<tr>
<td>InvalidBucketState</td>
<td>The request is not valid with the current state of the bucket.</td>
<td>409 Conflict</td>
<td>Client</td>
</tr>
<tr>
<td>InvalidDigest</td>
<td>The Content-MD5 you specified is not valid.</td>
<td>400 Bad Request</td>
<td>Client</td>
</tr>
<tr>
<td>InvalidEncryptionAlgorithmError</td>
<td>The encryption request that you specified is not valid. The valid value is AES256.</td>
<td>400 Bad Request</td>
<td>Client</td>
</tr>
<tr>
<td>Error Code</td>
<td>Description</td>
<td>HTTP Status Code</td>
<td>SOAP Fault Code Prefix</td>
</tr>
<tr>
<td>-----------------------------------</td>
<td>-----------------------------------------------------------------------------</td>
<td>------------------</td>
<td>------------------------</td>
</tr>
<tr>
<td>InvalidLocationConstraint</td>
<td>The specified location constraint is not valid. For more information about</td>
<td>400 Bad Request</td>
<td>Client</td>
</tr>
<tr>
<td></td>
<td>Regions, see <a href="#">How to Select a Region for Your Buckets</a>.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>InvalidObjectState</td>
<td>The operation is not valid for the current state of the object.</td>
<td>403 Forbidden</td>
<td>Client</td>
</tr>
<tr>
<td>InvalidPart</td>
<td>One or more of the specified parts could not be found. The part might not</td>
<td>400 Bad Request</td>
<td>Client</td>
</tr>
<tr>
<td></td>
<td>have been uploaded, or the specified entity tag might not have matched the</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>part's entity tag.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>InvalidPartOrder</td>
<td>The list of parts was not in ascending order. Parts list must be specified</td>
<td>400 Bad Request</td>
<td>Client</td>
</tr>
<tr>
<td></td>
<td>in order by part number.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>InvalidPayer</td>
<td>All access to this object has been disabled. For further assistance,</td>
<td>403 Forbidden</td>
<td>Client</td>
</tr>
<tr>
<td></td>
<td><a href="#">Contact Us</a>.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>InvalidPolicyDocument</td>
<td>The content of the form does not meet the conditions specified in the policy</td>
<td>400 Bad Request</td>
<td>Client</td>
</tr>
<tr>
<td></td>
<td>document.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>InvalidRange</td>
<td>The requested range cannot be satisfied.</td>
<td>416 Requested</td>
<td>Client</td>
</tr>
<tr>
<td></td>
<td>Requested Range Not Satisfiable.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Error Code</td>
<td>Description</td>
<td>HTTP Status Code</td>
<td>SOAP Fault Code Prefix</td>
</tr>
<tr>
<td>----------------</td>
<td>-----------------------------------------------------------------------------</td>
<td>------------------</td>
<td>------------------------</td>
</tr>
<tr>
<td>InvalidRequest</td>
<td>This error might occur for the following reasons:</td>
<td>400 Bad Request</td>
<td>Client</td>
</tr>
<tr>
<td></td>
<td>• Use AWS4-HMAC-SHA256.</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>• The access point can only be created for existing bucket.</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>• The access point is not in a state where it can be deleted.</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>• The access point can only be listed for an existing bucket.</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>• The next token is invalid.</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>• At least one action needs to be specified in a lifecycle rule.</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>• At least one lifecycle rule should be specified.</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>• Number of lifecycle rules should not exceed allowed limit of 1000 rules.</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>• Invalid range for parameter MaxResults.</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>• SOAP requests must be made over an HTTPS connection.</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>• Amazon S3 Transfer Acceleration is not supported for buckets with non-DNS compliant names.</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>• Amazon S3 Transfer Acceleration is not supported for buckets with periods (.) in their names.</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>• Amazon S3 Transfer Accelerate endpoint only supports virtual style requests.</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>• Amazon S3 Transfer Accelerate is not configured on this bucket.</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>• Amazon S3 Transfer Accelerate is disabled on this bucket.</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>• Amazon S3 Transfer Acceleration is not supported on this bucket. Contact AWS Support for more information.</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>• Amazon S3 Transfer Acceleration cannot be enabled on this bucket. Contact AWS Support for more information.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>InvalidSecurity</td>
<td>The provided security credentials are not valid.</td>
<td>403 Forbidden</td>
<td>Client</td>
</tr>
<tr>
<td>Error Code</td>
<td>Description</td>
<td>HTTP Status Code</td>
<td>SOAP Fault Code Prefix</td>
</tr>
<tr>
<td>-------------------------------</td>
<td>------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
<td>------------------</td>
<td>------------------------</td>
</tr>
<tr>
<td>InvalidSOAPRequest</td>
<td>The SOAP request body is invalid.</td>
<td>400 Bad Request</td>
<td>Client</td>
</tr>
<tr>
<td>InvalidStorageClass</td>
<td>The storage class you specified is not valid.</td>
<td>400 Bad Request</td>
<td>Client</td>
</tr>
<tr>
<td>InvalidTargetBucketForLogging</td>
<td>The target bucket for logging does not exist, is not owned by you, or does not have the appropriate grants for the log-delivery group.</td>
<td>400 Bad Request</td>
<td>Client</td>
</tr>
<tr>
<td>InvalidToken</td>
<td>The provided token is malformed or otherwise invalid.</td>
<td>400 Bad Request</td>
<td>Client</td>
</tr>
<tr>
<td>InvalidURI</td>
<td>Couldn't parse the specified URI.</td>
<td>400 Bad Request</td>
<td>Client</td>
</tr>
<tr>
<td>KeyTooLongError</td>
<td>Your key is too long.</td>
<td>400 Bad Request</td>
<td>Client</td>
</tr>
<tr>
<td>MalformedACLLError</td>
<td>The XML you provided was not well formed or did not validate against our published schema.</td>
<td>400 Bad Request</td>
<td>Client</td>
</tr>
<tr>
<td>MalformedPOSTRequest</td>
<td>The body of your POST request is not well-formed multipart/form-data.</td>
<td>400 Bad Request</td>
<td>Client</td>
</tr>
<tr>
<td>MalformedXML</td>
<td>This happens when the user sends malformed XML (XML that doesn't conform to the published XSD) for the configuration. The error message is, &quot;The XML you provided was not well formed or did not validate against our published schema.&quot;</td>
<td>400 Bad Request</td>
<td>Client</td>
</tr>
<tr>
<td>MaxMessageLengthExceeded</td>
<td>Your request was too big.</td>
<td>400 Bad Request</td>
<td>Client</td>
</tr>
<tr>
<td>MaxPostPreDataLengthExceededError</td>
<td>Your POST request fields preceding the upload file were too large.</td>
<td>400 Bad Request</td>
<td>Client</td>
</tr>
<tr>
<td>MetadataTooLarge</td>
<td>Your metadata headers exceed the maximum allowed metadata size.</td>
<td>400 Bad Request</td>
<td>Client</td>
</tr>
<tr>
<td>MethodNotAllowed</td>
<td>The specified method is not allowed against this resource.</td>
<td>405 Method Not Allowed</td>
<td>Client</td>
</tr>
<tr>
<td>MissingAttachment</td>
<td>A SOAP attachment was expected, but none were found.</td>
<td>N/A</td>
<td>Client</td>
</tr>
<tr>
<td>MissingContentLength</td>
<td>You must provide the Content-Length HTTP header.</td>
<td>411 Length Required</td>
<td>Client</td>
</tr>
<tr>
<td>Error Code</td>
<td>Description</td>
<td>HTTP Status Code</td>
<td>SOAP Fault Code Prefix</td>
</tr>
<tr>
<td>------------</td>
<td>-------------</td>
<td>------------------</td>
<td>------------------------</td>
</tr>
<tr>
<td>MissingRequestBodyError</td>
<td>This happens when the user sends an empty XML document as a request. The error message is, “Request body is empty.”</td>
<td>400 Bad Request</td>
<td>Client</td>
</tr>
<tr>
<td>MissingSecurityElement</td>
<td>The SOAP 1.1 request is missing a security element.</td>
<td>400 Bad Request</td>
<td>Client</td>
</tr>
<tr>
<td>MissingSecurityHeader</td>
<td>Your request is missing a required header.</td>
<td>400 Bad Request</td>
<td>Client</td>
</tr>
<tr>
<td>NoLoggingStatusForKey</td>
<td>There is no such thing as a logging status subresource for a key.</td>
<td>400 Bad Request</td>
<td>Client</td>
</tr>
<tr>
<td>NoSuchBucket</td>
<td>The specified bucket does not exist.</td>
<td>404 Not Found</td>
<td>Client</td>
</tr>
<tr>
<td>NoSuchBucketPolicy</td>
<td>The specified bucket does not have a bucket policy.</td>
<td>404 Not Found</td>
<td>Client</td>
</tr>
<tr>
<td>NoSuchKey</td>
<td>The specified key does not exist.</td>
<td>404 Not Found</td>
<td>Client</td>
</tr>
<tr>
<td>NoSuchLifecycleConfiguration</td>
<td>The lifecycle configuration does not exist.</td>
<td>404 Not Found</td>
<td>Client</td>
</tr>
<tr>
<td>NoSuchUpload</td>
<td>The specified multipart upload does not exist. The upload ID might be invalid, or the multipart upload might have been aborted or completed.</td>
<td>404 Not Found</td>
<td>Client</td>
</tr>
<tr>
<td>NoSuchVersion</td>
<td>Indicates that the version ID specified in the request does not match an existing version.</td>
<td>404 Not Found</td>
<td>Client</td>
</tr>
<tr>
<td>NotImplemented</td>
<td>A header you provided implies functionality that is not implemented.</td>
<td>501 Not Implemented</td>
<td>Server</td>
</tr>
<tr>
<td>NotSignedUp</td>
<td>Your account is not signed up for the Amazon S3 service. You must sign up before you can use Amazon S3. You can sign up at the following URL: <a href="https://aws.amazon.com/s3">https://aws.amazon.com/s3</a></td>
<td>403 Forbidden</td>
<td>Client</td>
</tr>
<tr>
<td>OperationAborted</td>
<td>A conflicting conditional operation is currently in progress against this resource. Try again.</td>
<td>409 Conflict</td>
<td>Client</td>
</tr>
<tr>
<td>PermanentRedirect</td>
<td>The bucket you are attempting to access must be addressed using the specified endpoint. Send all future requests to this endpoint.</td>
<td>301 Moved Permanently</td>
<td>Client</td>
</tr>
<tr>
<td>Error Code</td>
<td>Description</td>
<td>HTTP Status Code</td>
<td>SOAP Fault Code Prefix</td>
</tr>
<tr>
<td>------------------------</td>
<td>------------------------------------------------------------------------------</td>
<td>------------------</td>
<td>------------------------</td>
</tr>
<tr>
<td>PreconditionFailed</td>
<td>At least one of the preconditions you specified did not hold.</td>
<td>412</td>
<td>Client Precondition</td>
</tr>
<tr>
<td>Redirect</td>
<td>Temporary redirect.</td>
<td>307</td>
<td>Client</td>
</tr>
<tr>
<td>RestoreAlreadyInProgress</td>
<td>Object restore is already in progress.</td>
<td>409</td>
<td>Client Conflict</td>
</tr>
<tr>
<td>RequestIsNotMultiPartContent</td>
<td>Bucket POST must be of the enclosure-type multipart/form-data.</td>
<td>400</td>
<td>Client</td>
</tr>
<tr>
<td>RequestTimeout</td>
<td>Your socket connection to the server was not read from or written to within the timeout period.</td>
<td>400</td>
<td>Client</td>
</tr>
<tr>
<td>RequestTimeTooSkewed</td>
<td>The difference between the request time and the server's time is too large.</td>
<td>403</td>
<td>Client Forbidden</td>
</tr>
<tr>
<td>RequestTorrentOfBucketError</td>
<td>Requesting the torrent file of a bucket is not permitted.</td>
<td>400</td>
<td>Client</td>
</tr>
<tr>
<td>ServerSideEncryptionConfigurationNotServerSideEncryptionConfigurationWasFound</td>
<td>The server-side encryption configuration was not found.</td>
<td>400</td>
<td>Client</td>
</tr>
<tr>
<td>ServiceUnavailable</td>
<td>Reduce your request rate.</td>
<td>503</td>
<td>Server Service Unavailable</td>
</tr>
<tr>
<td>SignatureDoesNotMatch</td>
<td>The request signature that we calculated does not match the signature you provided. Check your AWS secret access key and signing method. For more information, see REST Authentication and SOAP Authentication.</td>
<td>403</td>
<td>Client Forbidden</td>
</tr>
<tr>
<td>SlowDown</td>
<td>Reduce your request rate.</td>
<td>503</td>
<td>Server Slow Down</td>
</tr>
<tr>
<td>TemporaryRedirect</td>
<td>You are being redirected to the bucket while DNS updates.</td>
<td>307</td>
<td>Client</td>
</tr>
<tr>
<td>TokenRefreshRequired</td>
<td>The provided token must be refreshed.</td>
<td>400</td>
<td>Client</td>
</tr>
<tr>
<td>TooManyAccessPoints</td>
<td>You have attempted to create more access point than allowed.</td>
<td>400</td>
<td>Client</td>
</tr>
<tr>
<td>TooManyBuckets</td>
<td>You have attempted to create more buckets than allowed.</td>
<td>400</td>
<td>Client</td>
</tr>
</tbody>
</table>
## List of SELECT Object Content Error Codes

The following table contains special errors that `SELECT Object Content` might return. For general information about Amazon S3 errors and a list of error codes, see [Error Responses](p. 816).

<table>
<thead>
<tr>
<th>Error Code</th>
<th>Description</th>
<th>HTTP Status Code</th>
<th>SOAP Fault Code Prefix</th>
</tr>
</thead>
<tbody>
<tr>
<td>UnexpectedContent</td>
<td>This request does not support content.</td>
<td>400 Bad Request</td>
<td>Client</td>
</tr>
<tr>
<td>UnresolvableGrantByEmailAddress</td>
<td>The email address you provided does not match any account on record.</td>
<td>400 Bad Request</td>
<td>Client</td>
</tr>
<tr>
<td>UserKeyMustBeSpecified</td>
<td>The bucket POST must contain the specified field name. If it is specified, check the order of the fields.</td>
<td>400 Bad Request</td>
<td>Client</td>
</tr>
<tr>
<td>NoSuchAccessPoint</td>
<td>The specified access point does not exist</td>
<td>400 Not Found</td>
<td>Client</td>
</tr>
<tr>
<td>InvalidTag</td>
<td>You have passed bad tag input - duplicate keys, key/values are too long, system tags were sent.</td>
<td>400 Bad Request</td>
<td>Client</td>
</tr>
<tr>
<td>MalformedPolicy</td>
<td>You have an invalid principal in policy.</td>
<td>400 Bad Request</td>
<td>Client</td>
</tr>
<tr>
<td>Error Code</td>
<td>Description</td>
<td>HTTP Status Code</td>
<td>SOAP Fault Code Prefix</td>
</tr>
<tr>
<td>-------------------------------</td>
<td>-----------------------------------------------------------------------------</td>
<td>------------------</td>
<td>------------------------</td>
</tr>
<tr>
<td>EmptyRequestBody</td>
<td>Request body cannot be empty.</td>
<td>400</td>
<td>Client</td>
</tr>
<tr>
<td>ExpressionTooLong</td>
<td>The SQL expression is too long: The maximum byte-length for the SQL expression is 256 KB.</td>
<td>400</td>
<td>Client</td>
</tr>
<tr>
<td>EvaluatorBindingDoesNotExist</td>
<td>A column name or a path provided does not exist in the SQL expression.</td>
<td>400</td>
<td>Client</td>
</tr>
<tr>
<td>EvaluatorInvalidArguments</td>
<td>Incorrect number of arguments in the function call in the SQL expression.</td>
<td>400</td>
<td>Client</td>
</tr>
<tr>
<td>EvaluatorInvalidTimestampFormatPattern</td>
<td>Invalid timestamp format string in the SQL expression.</td>
<td>400</td>
<td>Client</td>
</tr>
<tr>
<td>EvaluatorInvalidTimestampFormatPattern</td>
<td>Timestamp format pattern requires additional fields in the SQL expression.</td>
<td>400</td>
<td>Client</td>
</tr>
<tr>
<td>EvaluatorTimestampFormatPatternDuplicateFields</td>
<td>Timestamp format pattern contains multiple format specifiers representing the timestamp field in the SQL expression.</td>
<td>400</td>
<td>Client</td>
</tr>
<tr>
<td>EvaluatorTimestampFormatPatternHourClockAmPmMismatch</td>
<td>Timestamp format pattern contains a 12-hour hour of day format symbol but doesn't also contain an AM/PM field, or it contains a 24-hour hour of day format specifier and contains an AM/PM field in the SQL expression.</td>
<td>400</td>
<td>Client</td>
</tr>
<tr>
<td>EvaluatorInvalidTimestampFormatPatternSymbolForParsing</td>
<td>Timestamp format pattern contains a valid format symbol that cannot be applied to timestamp parsing in the SQL expression.</td>
<td>400</td>
<td>Client</td>
</tr>
<tr>
<td>EvaluatorUnterminatedTimestampFormatPatternToken</td>
<td>Timestamp format pattern contains unterminated token in the SQL expression.</td>
<td>400</td>
<td>Client</td>
</tr>
<tr>
<td>EvaluatorInvalidTimestampFormatPatternToken</td>
<td>Timestamp format pattern contains an invalid token in the SQL expression.</td>
<td>400</td>
<td>Client</td>
</tr>
<tr>
<td>EvaluatorInvalidTimestampFormatPatternSymbol</td>
<td>Timestamp format pattern contains an invalid symbol in the SQL expression.</td>
<td>400</td>
<td>Client</td>
</tr>
<tr>
<td>IllegalSqlFunctionArgument</td>
<td>Illegal argument was used in the SQL function.</td>
<td>400</td>
<td>Client</td>
</tr>
<tr>
<td>InternalError</td>
<td>Encountered an internal error.</td>
<td>500</td>
<td>Client</td>
</tr>
<tr>
<td>Error Code</td>
<td>Description</td>
<td>HTTP Status Code</td>
<td>SOAP Fault Code Prefix</td>
</tr>
<tr>
<td>------------------------------</td>
<td>--------------------------------------------------------------------</td>
<td>------------------</td>
<td>------------------------</td>
</tr>
<tr>
<td>InvalidColumnIndex</td>
<td>Column index in the SQL expression is invalid.</td>
<td>400</td>
<td>Client</td>
</tr>
<tr>
<td>InvalidCompressionFormat</td>
<td>The file is not in a supported compression format. Only GZIP and BZIP2 are supported.</td>
<td>400</td>
<td>Client</td>
</tr>
<tr>
<td>InvalidExpressionType</td>
<td>The ExpressionType is invalid. Only SQL expressions are supported.</td>
<td>400</td>
<td>Client</td>
</tr>
<tr>
<td>InvalidFileHeaderInfo</td>
<td>The FileHeaderInfo is invalid. Only NONE, USE, and IGNORE are supported.</td>
<td>400</td>
<td>Client</td>
</tr>
<tr>
<td>InvalidKeyPath</td>
<td>Key path in the SQL expression is invalid.</td>
<td>400</td>
<td>Client</td>
</tr>
<tr>
<td>InvalidJsonType</td>
<td>The JsonType is invalid. Only DOCUMENT and LINES are supported.</td>
<td>400</td>
<td>Client</td>
</tr>
<tr>
<td>InvalidQuoteFields</td>
<td>The QuoteFields is invalid. Only ALWAYS and ASNEEDED are supported.</td>
<td>400</td>
<td>Client</td>
</tr>
<tr>
<td>InvalidRequestParameter</td>
<td>The value of a parameter in SelectRequest element is invalid. Check the service API documentation and try again.</td>
<td>400</td>
<td>Client</td>
</tr>
<tr>
<td>OverMaxColumn</td>
<td>The number of columns in the result is greater than the maximum allowable number of columns.</td>
<td>400</td>
<td>Client</td>
</tr>
<tr>
<td>OverMaxRecordSize</td>
<td>The length of a record in the input or result is greater than maxCharsPerRecord of 1 MB.</td>
<td>400</td>
<td>Client</td>
</tr>
<tr>
<td>TruncatedInput</td>
<td>Object decompression failed. Check that the object is properly compressed using the format specified in the request.</td>
<td>400</td>
<td>Client</td>
</tr>
<tr>
<td>UnauthorizedAccess</td>
<td>You are not authorized to perform this operation.</td>
<td>401</td>
<td>Client</td>
</tr>
<tr>
<td>ExternalEvalException</td>
<td>The query cannot be evaluated. Check the file and try again.</td>
<td>400</td>
<td>Client</td>
</tr>
<tr>
<td>InvalidDataSource</td>
<td>Invalid data source type. Only CSV, JSON, and Parquet are supported.</td>
<td>400</td>
<td>Client</td>
</tr>
<tr>
<td>InvalidDataType</td>
<td>The SQL expression contains an invalid data type.</td>
<td>400</td>
<td>Client</td>
</tr>
<tr>
<td>InvalidTableAlias</td>
<td>The SQL expression contains an invalid table alias.</td>
<td>400</td>
<td>Client</td>
</tr>
</tbody>
</table>
## List of SELECT Object Content Error Codes

<table>
<thead>
<tr>
<th>Error Code</th>
<th>Description</th>
<th>HTTP Status Code</th>
<th>SOAP Fault Code Prefix</th>
</tr>
</thead>
<tbody>
<tr>
<td>InvalidTextEncoding</td>
<td>Invalid encoding type. Only UTF-8 encoding is supported.</td>
<td>400</td>
<td>Client</td>
</tr>
<tr>
<td>JSONParsingError</td>
<td>Encountered an error parsing the JSON file. Check the file and try again.</td>
<td>400</td>
<td>Client</td>
</tr>
<tr>
<td>UnrecognizedFormatException</td>
<td>Encountered an invalid record type.</td>
<td>400</td>
<td>Client</td>
</tr>
<tr>
<td>MalformedXML</td>
<td>The XML provided was not well formed or did not validate against our published schema. Check the service documentation and try again.</td>
<td>400</td>
<td>Client</td>
</tr>
<tr>
<td>MissingRequiredParameter</td>
<td>The SelectRequest entity is missing a required parameter. Check the service documentation and try again.</td>
<td>400</td>
<td>Client</td>
</tr>
<tr>
<td>MultipleDataSourcesUnsupported</td>
<td>Multiple data sources are not supported.</td>
<td>400</td>
<td>Client</td>
</tr>
<tr>
<td>ObjectSerializationConflict</td>
<td>InputSerialization specifies more than one format (CSV, JSON, or Parquet), or OutputSerialization specifies more than one format (CSV or JSON). InputSerialization and OutputSerialization can only specify one format each.</td>
<td>400</td>
<td>Client</td>
</tr>
<tr>
<td>UnsupportedFunction</td>
<td>Encountered an unsupported SQL function.</td>
<td>400</td>
<td>Client</td>
</tr>
<tr>
<td>UnsupportedSqlOperation</td>
<td>Encountered an unsupported SQL operation.</td>
<td>400</td>
<td>Client</td>
</tr>
<tr>
<td>UnsupportedSqlStructure</td>
<td>Encountered an unsupported SQL structure. Check the SQL Reference.</td>
<td>400</td>
<td>Client</td>
</tr>
<tr>
<td>UnsupportedStorageClass</td>
<td>Encountered an invalid storage class. Only STANDARD, STANDARD_IA, and ONEZONE_IA storage classes are supported.</td>
<td>400</td>
<td>Client</td>
</tr>
<tr>
<td>UnsupportedSyntax</td>
<td>Encountered invalid syntax.</td>
<td>400</td>
<td>Client</td>
</tr>
<tr>
<td>UnsupportedRangeHeader</td>
<td>Range header is not supported for this operation.</td>
<td>400</td>
<td>Client</td>
</tr>
<tr>
<td>LexerInvalidChar</td>
<td>The SQL expression contains an invalid character.</td>
<td>400</td>
<td>Client</td>
</tr>
<tr>
<td>LexerInvalidOperator</td>
<td>The SQL expression contains an invalid literal.</td>
<td>400</td>
<td>Client</td>
</tr>
</tbody>
</table>
## List of SELECT Object Content Error Codes

<table>
<thead>
<tr>
<th>Error Code</th>
<th>Description</th>
<th>HTTP Status Code</th>
<th>SOAP Fault Code Prefix</th>
</tr>
</thead>
<tbody>
<tr>
<td>LexerInvalidLiteral</td>
<td>The SQL expression contains an invalid operator.</td>
<td>400</td>
<td>Client</td>
</tr>
<tr>
<td>LexerInvalidIONLiteral</td>
<td>The SQL expression contains an invalid operator.</td>
<td>400</td>
<td>Client</td>
</tr>
<tr>
<td>ParseExpectedDatePart</td>
<td>Did not find the expected date part in the SQL expression.</td>
<td>400</td>
<td>Client</td>
</tr>
<tr>
<td>ParseExpectedKeyword</td>
<td>Did not find the expected keyword in the SQL expression.</td>
<td>400</td>
<td>Client</td>
</tr>
<tr>
<td>ParseExpectedTokenType</td>
<td>Did not find the expected token in the SQL expression.</td>
<td>400</td>
<td>Client</td>
</tr>
<tr>
<td>ParseExpected2TokenTypes</td>
<td>Did not find the expected token in the SQL expression.</td>
<td>400</td>
<td>Client</td>
</tr>
<tr>
<td>ParseExpectedNumber</td>
<td>Did not find the expected number in the SQL expression.</td>
<td>400</td>
<td>Client</td>
</tr>
<tr>
<td>ParseExpectedRightParenBuiltinFunctionCall</td>
<td>Did not find the expected right parenthesis character in the SQL expression.</td>
<td>400</td>
<td>Client</td>
</tr>
<tr>
<td>ParseExpectedTypeName</td>
<td>Did not find the expected type name in the SQL expression.</td>
<td>400</td>
<td>Client</td>
</tr>
<tr>
<td>ParseExpectedWhenClause</td>
<td>Did not find the expected WHEN clause in the SQL expression. CASE is not supported.</td>
<td>400</td>
<td>Client</td>
</tr>
<tr>
<td>ParseUnsupportedToken</td>
<td>The SQL expression contains an unsupported token.</td>
<td>400</td>
<td>Client</td>
</tr>
<tr>
<td>ParseUnsupportedLiteralsGroupBy</td>
<td>The SQL expression contains an unsupported use of GROUP BY.</td>
<td>400</td>
<td>Client</td>
</tr>
<tr>
<td>ParseExpectedMember</td>
<td>The SQL expression contains an unsupported use of MEMBER.</td>
<td>400</td>
<td>Client</td>
</tr>
<tr>
<td>ParseUnsupportedSelect</td>
<td>The SQL expression contains an unsupported use of SELECT.</td>
<td>400</td>
<td>Client</td>
</tr>
<tr>
<td>ParseUnsupportedCase</td>
<td>The SQL expression contains an unsupported use of CASE.</td>
<td>400</td>
<td>Client</td>
</tr>
<tr>
<td>ParseUnsupportedCaseClause</td>
<td>The SQL expression contains an unsupported use of CASE.</td>
<td>400</td>
<td>Client</td>
</tr>
<tr>
<td>ParseUnsupportedAlias</td>
<td>The SQL expression contains an unsupported use of ALIAS.</td>
<td>400</td>
<td>Client</td>
</tr>
<tr>
<td>ParseUnsupportedSyntax</td>
<td>The SQL expression contains unsupported syntax.</td>
<td>400</td>
<td>Client</td>
</tr>
<tr>
<td>Error Code</td>
<td>Description</td>
<td>HTTP Status Code</td>
<td>SOAP Fault Code Prefix</td>
</tr>
<tr>
<td>-------------------------------</td>
<td>-----------------------------------------------------------------------------</td>
<td>------------------</td>
<td>------------------------</td>
</tr>
<tr>
<td>ParseUnknownOperator</td>
<td>The SQL expression contains an invalid operator.</td>
<td>400</td>
<td>Client</td>
</tr>
<tr>
<td>ParseInvalidPathComponent</td>
<td>The SQL expression contains an invalid path component.</td>
<td>400</td>
<td>Client</td>
</tr>
<tr>
<td>ParseMissingIdentAfterAt</td>
<td>Did not find the expected identifier after the @ symbol in the SQL expression.</td>
<td>400</td>
<td>Client</td>
</tr>
<tr>
<td>ParseUnexpectedOperator</td>
<td>The SQL expression contains an unexpected operator.</td>
<td>400</td>
<td>Client</td>
</tr>
<tr>
<td>ParseUnexpectedTerm</td>
<td>The SQL expression contains an unexpected term.</td>
<td>400</td>
<td>Client</td>
</tr>
<tr>
<td>ParseUnexpectedToken</td>
<td>The SQL expression contains an unexpected token.</td>
<td>400</td>
<td>Client</td>
</tr>
<tr>
<td>ParseUnExpectedKeyword</td>
<td>The SQL expression contains an unexpected keyword.</td>
<td>400</td>
<td>Client</td>
</tr>
<tr>
<td>ParseExpectedExpression</td>
<td>Did not find the expected SQL expression.</td>
<td>400</td>
<td>Client</td>
</tr>
<tr>
<td>ParseExpectedLeftParenAfterCast</td>
<td>Did not find the expected left parenthesis after CAST in the SQL expression.</td>
<td>400</td>
<td>Client</td>
</tr>
<tr>
<td>ParseExpectedLeftParenValueConstructor</td>
<td>Did not find the expected left parenthesis in the SQL expression.</td>
<td>400</td>
<td>Client</td>
</tr>
<tr>
<td>ParseExpectedLeftParenBuiltinFunctionCall</td>
<td>Did not find the expected left parenthesis in the SQL expression.</td>
<td>400</td>
<td>Client</td>
</tr>
<tr>
<td>ParseExpectedArgumentDelimiter</td>
<td>Did not find the expected argument delimiter in the SQL expression.</td>
<td>400</td>
<td>Client</td>
</tr>
<tr>
<td>ParseCastArity</td>
<td>The SQL expression CAST has incorrect arity.</td>
<td>400</td>
<td>Client</td>
</tr>
<tr>
<td>ParseInvalidTypeParam</td>
<td>The SQL expression contains an invalid parameter value.</td>
<td>400</td>
<td>Client</td>
</tr>
<tr>
<td>ParseEmptySelect</td>
<td>The SQL expression contains an empty SELECT.</td>
<td>400</td>
<td>Client</td>
</tr>
<tr>
<td>ParseSelectMissingFrom</td>
<td>The SQL expression contains a missing FROM after SELECT list.</td>
<td>400</td>
<td>Client</td>
</tr>
<tr>
<td>ParseExpectedIdentForGroup</td>
<td>GROUP is not supported in the SQL expression.</td>
<td>400</td>
<td>Client</td>
</tr>
<tr>
<td>ParseExpectedIdentForAlias</td>
<td>Did not find the expected identifier for the alias in the SQL expression.</td>
<td>400</td>
<td>Client</td>
</tr>
<tr>
<td>Error Code</td>
<td>Description</td>
<td>HTTP Status Code</td>
<td>SOAP Fault Code Prefix</td>
</tr>
<tr>
<td>------------------------------------</td>
<td>-----------------------------------------------------------------------------</td>
<td>------------------</td>
<td>------------------------</td>
</tr>
<tr>
<td>ParseUnsupportedCallWithStar</td>
<td>Only COUNT with (*) as a parameter is supported in the SQL expression.</td>
<td>400</td>
<td>Client</td>
</tr>
<tr>
<td>ParseNonUnaryAggregateFunctionCall</td>
<td>Only one argument is supported for aggregate functions in the SQL expression.</td>
<td>400</td>
<td>Client</td>
</tr>
<tr>
<td>ParseMalformedJoin</td>
<td>JOIN is not supported in the SQL expression.</td>
<td>400</td>
<td>Client</td>
</tr>
<tr>
<td>ParseExpectedIdentForAt</td>
<td>Did not find the expected identifier for AT name in the SQL expression.</td>
<td>400</td>
<td>Client</td>
</tr>
<tr>
<td>ParseAsteriskIsNotAloneInSelectList</td>
<td>Other expressions are not allowed in the SELECT list when '*' is used without dot notation in the SQL expression.</td>
<td>400</td>
<td>Client</td>
</tr>
<tr>
<td>ParseCannotMixSqbAndWildcardInSelectList</td>
<td>Cannot mix [ ] and * in the same expression in a SELECT list in SQL expression.</td>
<td>400</td>
<td>Client</td>
</tr>
<tr>
<td>ParseInvalidContextForWildcardInSelectList</td>
<td>Invalid use of * in SELECT list in the SQL expression.</td>
<td>400</td>
<td>Client</td>
</tr>
<tr>
<td>ValueParseFailure</td>
<td>Timestamp parse failure in the SQL expression.</td>
<td>400</td>
<td>Client</td>
</tr>
<tr>
<td>IncorrectSqlFunctionArgumentType</td>
<td>Incorrect type of arguments in function call in the SQL expression.</td>
<td>400</td>
<td>Client</td>
</tr>
<tr>
<td>AmbiguousFieldName</td>
<td>Field name matches to multiple fields in the file. Check the SQL expression and the file, and try again.</td>
<td>400</td>
<td>Client</td>
</tr>
<tr>
<td>ValueParseFailure</td>
<td>Timestamp parse failure in the SQL expression.</td>
<td>400</td>
<td>Client</td>
</tr>
<tr>
<td>IntegerOverflow</td>
<td>Integer overflow or underflow in the SQL expression.</td>
<td>400</td>
<td>Client</td>
</tr>
<tr>
<td>LikeInvalidInputs</td>
<td>Invalid argument given to the LIKE clause in the SQL expression.</td>
<td>400</td>
<td>Client</td>
</tr>
<tr>
<td>InvalidCast</td>
<td>Attempt to convert from one data type to another using CAST failed in the SQL expression.</td>
<td>400</td>
<td>Client</td>
</tr>
<tr>
<td>ParquetParsingError</td>
<td>Error parsing Parquet file. Please check the file and try again.</td>
<td>400</td>
<td>Client</td>
</tr>
<tr>
<td>NumberFormatError</td>
<td>Error parsing a number. This can be caused by under/over flow of integers.</td>
<td>400</td>
<td>Client</td>
</tr>
</tbody>
</table>
The following table contains special errors that the Replication operation might return. For general information about Amazon S3 errors and a list of error codes, see Error Responses (p. 816).

<table>
<thead>
<tr>
<th>Error Code</th>
<th>Description</th>
<th>HTTP Status Code</th>
<th>SOAP Fault Code Prefix</th>
</tr>
</thead>
<tbody>
<tr>
<td>InvalidArgument</td>
<td>This error might occur for the following reasons:</td>
<td>400</td>
<td>Client</td>
</tr>
<tr>
<td></td>
<td>• The &lt;Account&gt; element is empty. It must contain a valid account ID.</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>• The AWS account specified in the &lt;Account&gt; element must match the destination bucket owner.</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>• ReplicationTime-Status must contain a value.</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>• ReplicationTime-ReplicationTimeValue must contain a value.</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>• Replication-ReplicationTimeValue-Minutes value must be 15.</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>• ReplicationMetrics must contain a Status.</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>• ReplicationMetrics must contain an EventThreshold.</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>• EventThreshold-ReplicationTimeValue-Minutes value must be 15.</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>• Rule ID must not contain non-ASCII characters.</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

API Version 2006-03-01
832
### List of Amazon S3 on Outposts Error Codes

The following table contains special errors that Amazon S3 on Outposts operation might return. For general information about general Amazon S3 errors and a list of error codes, see Error Responses (p. 816).

<table>
<thead>
<tr>
<th>Error Code</th>
<th>Description</th>
<th>HTTP Status Code</th>
<th>SOAP Fault Code Prefix</th>
</tr>
</thead>
<tbody>
<tr>
<td>InvalidRequest</td>
<td>This error might occur for the following reasons:</td>
<td>400</td>
<td>Client</td>
</tr>
<tr>
<td></td>
<td>• The <code>&lt;Owner&gt;</code> in <code>&lt;AccessControlTranslation&gt;</code> has a value, so the <code>&lt;Account&gt;</code> element must be specified.</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>• The <code>&lt;Account&gt;</code> element is empty. It must contain a valid account ID.</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>• Replication destination must contain both <code>ReplicationTime</code> and <code>Metrics</code> or neither.</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>• <code>ReplicationTime</code> and <code>ReplicationMetrics</code> should have the same status.</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>• S3 Replication Time Control (S3 RTC) is not supported in this AWS Region.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>NoSuchReplicationConfiguration</td>
<td>There is no replication configuration with that name.</td>
<td>400</td>
<td>Client</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Error Code</th>
<th>Description</th>
<th>HTTP Status Code</th>
<th>SOAP Fault Code Prefix</th>
</tr>
</thead>
<tbody>
<tr>
<td>BadRequest</td>
<td>The bucket is in transitional state due to a previous deletion attempt; try again later.</td>
<td>400</td>
<td>Bad Request</td>
</tr>
<tr>
<td></td>
<td>This error might occur for the following reasons:</td>
<td></td>
<td>Not supported</td>
</tr>
<tr>
<td></td>
<td>• Amazon VPC configuration is required.</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>• Public Access is not allowed on S3 on Outposts access point.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>InvalidRequest</td>
<td>The request is not valid with the current state of the Outpost.</td>
<td>409</td>
<td>Conflict</td>
</tr>
<tr>
<td>InvalidOutpostState</td>
<td>Access Point is not in a state where it can be deleted.</td>
<td>400</td>
<td>Bad Request</td>
</tr>
<tr>
<td>NoSuchOutpost</td>
<td>This error occurs when the Outpost specified does not exist.</td>
<td>404</td>
<td>Not Found</td>
</tr>
</tbody>
</table>

API Version 2006-03-01
833
<table>
<thead>
<tr>
<th>Error Code</th>
<th>Description</th>
<th>HTTP Status Code</th>
<th>SOAP Fault Code Prefix</th>
</tr>
</thead>
<tbody>
<tr>
<td>NoSuchTagSet</td>
<td>This error occurs when the Outpost specified does not exist.</td>
<td>404 Not Found</td>
<td>Not supported</td>
</tr>
<tr>
<td>UnsupportedOperation</td>
<td>This specified action was not supported.</td>
<td>404 Not Found</td>
<td>Not supported</td>
</tr>
</tbody>
</table>
AWS glossary

For the latest AWS terminology, see the AWS glossary in the AWS General Reference.
Amazon S3 Resources

Following is a table that lists related resources that you'll find useful as you work with this service.

<table>
<thead>
<tr>
<th>Resource</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Amazon Simple Storage Service Getting Started Guide</td>
<td>The getting started guide provides a quick tutorial of the service based on a simple use case.</td>
</tr>
<tr>
<td>Amazon Simple Storage Service Developer Guide</td>
<td>The developer guide describes how to accomplish tasks using Amazon S3 operations.</td>
</tr>
<tr>
<td>Amazon S3 Technical FAQ</td>
<td>The FAQ covers the top 20 questions developers have asked about this product.</td>
</tr>
<tr>
<td>Amazon S3 Release Notes</td>
<td>The Release Notes give a high-level overview of the current release. They specifically note any new features, corrections, and known issues.</td>
</tr>
<tr>
<td>Tools for Amazon Web Services</td>
<td>A central starting point to find documentation, code samples, release notes, and other information to help you build innovative applications with AWS SDKs and tools.</td>
</tr>
<tr>
<td>AWS Management Console</td>
<td>The console allows you to perform most of the functions of Amazon S3 without programming.</td>
</tr>
<tr>
<td>Discussion Forums</td>
<td>A community-based forum for developers to discuss technical questions related to Amazon Web Services.</td>
</tr>
<tr>
<td>AWS Support Center</td>
<td>The home page for AWS Technical Support, including access to our Developer Forums, Technical FAQs, Service Status page, and Premium Support.</td>
</tr>
<tr>
<td>AWS Premium Support</td>
<td>The primary web page for information about AWS Premium Support, a one-on-one, fast-response support channel to help you build and run applications on AWS Infrastructure Services.</td>
</tr>
<tr>
<td>Amazon S3 product information</td>
<td>The primary web page for information about Amazon S3.</td>
</tr>
<tr>
<td>Contact Us</td>
<td>A central contact point for inquiries concerning AWS billing, account, events, abuse, etc.</td>
</tr>
<tr>
<td>Conditions of Use</td>
<td>Detailed information about the copyright and trademark usage at Amazon.com and other topics.</td>
</tr>
</tbody>
</table>
## Document History

The following table describes the important changes in each release of the Amazon Simple Storage Service API Reference up to March 27, 2019. For changes after March 27, 2019, see the consolidated Document History in the Amazon Simple Storage Service Developer Guide.

- **API version:** 2006-03-01
- **Latest documentation update:** March 27, 2019

<table>
<thead>
<tr>
<th>Change</th>
<th>Description</th>
<th>Release Date</th>
</tr>
</thead>
<tbody>
<tr>
<td>New archive storage class</td>
<td>Amazon S3 now offers a new archive storage class, DEEP_ARCHIVE, for storing rarely accessed objects. For more information, see Storage Classes in the Amazon Simple Storage Service Developer Guide.</td>
<td>March 27, 2019</td>
</tr>
</tbody>
</table>
| Support for Parquet-formatted Amazon S3 inventory files | Amazon S3 now supports the Apache Parquet (Parquet) format in addition to the Apache optimized row columnar (ORC) and comma-separated values (CSV) file formats for inventory output files. For more information, see Amazon S3 Inventory in the Amazon Simple Storage Service Developer Guide. The following APIs were updated accordingly:  
  - GetBucketInventoryConfiguration (p. 106)  
  - PutBucketInventoryConfiguration (p. 281) | December 04, 2018 |
| PUT directly to the GLACIER storage class | The Amazon S3 PUT and related operations now support specifying GLACIER as the storage class when creating objects. Previously, you had to transition to the GLACIER storage class from another Amazon S3 storage class. For more information about the GLACIER storage class, see Storage Classes in the Amazon Simple Storage Service Developer Guide. The following APIs were updated accordingly:  
  - PutObject (p. 343)  
  - POST Object (p. 769)  
  - CopyObject (p. 17)  
  - CreateMultipartUpload (p. 33) | November 26, 2018 |
| Object Lock | Amazon S3 now supports locking objects using a Write Once Read Many (WORM) model. You can lock objects for a definite period of time using a retention period or indefinitely using a legal hold. For more information about Amazon S3 Object Lock, see Locking Objects in the Amazon Simple Storage Service Developer Guide. The following APIs were updated for S3 Object Lock:  
  - PutObject (p. 343) | November 26, 2018 |
<table>
<thead>
<tr>
<th>Change</th>
<th>Description</th>
<th>Release Date</th>
</tr>
</thead>
<tbody>
<tr>
<td>New storage class</td>
<td>Amazon S3 now offers a new storage class named INTELLIGENT_TIERING that is for storing data that has changing or unknown access patterns. For more information, see Storage Classes in the Amazon Simple Storage Service Developer Guide. The following APIs were updated accordingly:</td>
<td>November 26, 2018</td>
</tr>
<tr>
<td>• PutObject (p. 343)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>• POST Object (p. 769)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>• CopyObject (p. 17)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>• CreateMultipartUpload (p. 33)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Block Public Access</td>
<td>Amazon S3 now includes the ability to block public access to buckets and objects on a per-bucket or account-wide basis. For more information, see Using Amazon S3 Block Public Access in the Amazon Simple Storage Service Developer Guide.</td>
<td>November 15, 2018</td>
</tr>
<tr>
<td>Filtering enhancements in cross-region replication (CRR) rules</td>
<td>In a CRR rule configuration, you can specify an object filter to choose a subset of objects to apply the rule to. Previously, you could filter only on an object key prefix. In this release, you can filter on an object key prefix, one or more object tags, or both. For more information, see Replication Configuration Overview in the Amazon Simple Storage Service Developer Guide. The following APIs are updated accordingly:</td>
<td>September 19, 2018</td>
</tr>
<tr>
<td>• PutBucketReplication (p. 320)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>• GetBucketReplication (p. 139)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>• DeleteBucketReplication (p. 63)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>New storage class</td>
<td>Amazon S3 now offers a new storage class, ONEZONE_IA (IA, for infrequent access) for storing objects. For more information, see Storage Classes in the Amazon Simple Storage Service Developer Guide.</td>
<td>April 4, 2018</td>
</tr>
<tr>
<td>Amazon S3 Select</td>
<td>Amazon S3 Select is now generally available. This feature retrieves object content based on an SQL expression. For more information, see Selecting Content from Objects in the Amazon Simple Storage Service Developer Guide. The following API has been updated:</td>
<td>April 4, 2018</td>
</tr>
<tr>
<td>• SelectObjectContent (p. 387)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Change</td>
<td>Description</td>
<td>Release Date</td>
</tr>
<tr>
<td>--------</td>
<td>-------------</td>
<td>-------------</td>
</tr>
<tr>
<td>Asia Pacific (Osaka-Local) Region</td>
<td>Amazon S3 is now available in the Asia Pacific (Osaka-Local) Region. For more information about Amazon S3 Regions and endpoints, see Regions and Endpoints in the AWS General Reference. <strong>Important</strong> You can use the Asia Pacific (Osaka-Local) Region only in conjunction with the Asia Pacific (Tokyo) Region. To request access to Asia Pacific (Osaka-Local) Region, contact your sales representative.</td>
<td>February 12, 2018</td>
</tr>
<tr>
<td>Europe (Paris) Region</td>
<td>Amazon S3 is now available in the Europe (Paris) Region. For more information about Amazon S3 regions and endpoints, see Regions and Endpoints in the AWS General Reference.</td>
<td>December 18, 2017</td>
</tr>
<tr>
<td>China (Ningxia) Region</td>
<td>Amazon S3 is now available in the China (Ningxia) Region. For more information about Amazon S3 regions and endpoints, see Regions and Endpoints in the AWS General Reference.</td>
<td>December 11, 2017</td>
</tr>
<tr>
<td>Querying archives with SQL</td>
<td>Amazon S3 now supports querying S3 Glacier data archives with SQL. For more information, see Querying Archived Objects in the Amazon Simple Storage Service Developer Guide. The following API changed: • RestoreObject (p. 378)</td>
<td>November 29, 2017</td>
</tr>
<tr>
<td>SELECT Object Content (Preview)</td>
<td>Amazon S3 now supports the SELECT Object Content functionality as part of a Preview program. This feature retrieves object content based on an SQL expression. The following API has been added: • SelectObjectContent (p. 387)</td>
<td>November 29, 2017</td>
</tr>
<tr>
<td>Support for ORC-formatted Amazon S3 inventory files</td>
<td>Amazon S3 now supports the Apache optimized row columnar (ORC) format in addition to comma-separated values (CSV) file format for inventory output files. For more information, see Amazon S3 Inventory in the Amazon Simple Storage Service Developer Guide. The following APIs are updated accordingly: • GetBucketInventoryConfiguration (p. 106) • PutBucketInventoryConfiguration (p. 281)</td>
<td>November 17, 2017</td>
</tr>
<tr>
<td>Change</td>
<td>Description</td>
<td>Release Date</td>
</tr>
<tr>
<td>--------</td>
<td>-------------</td>
<td>--------------</td>
</tr>
</tbody>
</table>
| Default encryption for S3 buckets | Amazon S3 default encryption provides a way to set the default encryption behavior for an S3 bucket. You can set default encryption on a bucket so that all objects are encrypted when they are stored in the bucket. The objects are encrypted using server-side encryption with either Amazon S3-managed keys (SSE-S3) or AWS KMS-managed keys (SSE-KMS). For more information, see Amazon S3 Default Encryption for S3 Buckets in the Amazon Simple Storage Service Developer Guide. The following APIs are updated accordingly:  
  - DeleteBucketEncryption (p. 49)  
  - GetBucketEncryption (p. 100)  
  - PutBucketEncryption (p. 275) | November 06, 2017 |
| Encryption status in Amazon S3 inventory | Amazon S3 now supports including encryption status in Amazon S3 inventory so you can see how your objects are encrypted at rest for compliance auditing or other purposes. You can also configure to encrypt Amazon S3 inventory with server-side encryption (SSE) or SSE-KMS so that all inventory files are encrypted accordingly. For more information, see Amazon S3 Inventory in the Amazon Simple Storage Service Developer Guide. The following APIs are updated accordingly:  
  - GetBucketInventoryConfiguration (p. 106)  
  - PutBucketInventoryConfiguration (p. 281) | November 06, 2017 |
| Cross-region replication (CRR) enhancements | Cross-region replication (CRR) now supports the following:  
  - In a cross-account scenario, you can add a CRR configuration to change replica ownership to the AWS account that owns the destination bucket. For more information, see CRR: Change Replica Owner in the Amazon Simple Storage Service Developer Guide.  
  - By default, Amazon S3 does not replicate objects in your source bucket that are created using server-side encryption using AWS KMS-managed keys. In your CRR configuration, you can now direct Amazon S3 to replicate these objects. For more information, see CRR: Replicating Objects Created with SEE Using AWS KMS-Managed Encryption Keys in the Amazon Simple Storage Service Developer Guide. The following APIs are updated accordingly:  
  - GetBucketReplication (p. 139)  
  - PutBucketReplication (p. 320) | November 06, 2017 |
<table>
<thead>
<tr>
<th>Change</th>
<th>Description</th>
<th>Release Date</th>
</tr>
</thead>
<tbody>
<tr>
<td>Europe (London) Region</td>
<td>Amazon S3 is now available in the Europe (London) Region. For more information about Amazon S3 regions and endpoints, see Regions and Endpoints in the AWS General Reference.</td>
<td>December 13, 2016</td>
</tr>
<tr>
<td>Canada (Central) Region</td>
<td>Amazon S3 is now available in the Canada (Central) Region. For more information about Amazon S3 regions and endpoints, see Regions and Endpoints in the AWS General Reference.</td>
<td>December 8, 2016</td>
</tr>
</tbody>
</table>
| Object tagging support | Amazon S3 now supports object tagging. The following new API operations support object tagging:  
  - PutObjectTagging (p. 370)  
  - GetObjectTagging (p. 176)  
  - DeleteObjectTagging (p. 82)  
  In addition, other API operations are updated to support object tagging. For more information, see Object Tagging in the Amazon Simple Storage Service Developer Guide. | November 29, 2016 |
| S3 lifecycle now supports object tag based filter | Amazon S3 now supports tag-based filtering in lifecycle configuration. You can now specify a lifecycle rule, in which you can specify a key prefix, one or more object tags, or a combination of both, to select a subset of objects to which the lifecycle rule applies. For more information, see Object Lifecycle Management in the Amazon Simple Storage Service Developer Guide.  
  Amazon S3 now supports Expedited and Bulk data retrievals in addition to Standard retrievals when restoring objects archived to S3 Glacier. | November 29, 2016 |
| CloudWatch request metrics for buckets | Amazon S3 now supports CloudWatch metrics for requests made on buckets. The following new API operations support configuring request metrics:  
  - DeleteBucketMetricsConfiguration (p. 57)  
  - GetBucketMetricsConfiguration (p. 122)  
  - PutBucketMetricsConfiguration (p. 302)  
  - ListBucketMetricsConfigurations (p. 208)  
  For more information, see Monitoring Metrics with Amazon CloudWatch in the Amazon Simple Storage Service Developer Guide. | November 29, 2016 |
<table>
<thead>
<tr>
<th>Change</th>
<th>Description</th>
<th>Release Date</th>
</tr>
</thead>
</table>
| Amazon S3 Inventory | Amazon S3 now supports storage inventory. Amazon S3 inventory provides a flat-file output of your objects and their corresponding metadata on a daily or weekly basis for an S3 bucket or a shared prefix (that is, objects that have names that begin with a common string). The following new API operations are for storage inventory:  
  - `DeleteBucketInventoryConfiguration` (p. 53)  
  - `GetBucketInventoryConfiguration` (p. 106)  
  - `PutBucketInventoryConfiguration` (p. 281)  
  - `ListBucketInventoryConfigurations` (p. 203)  
  For more information, see Amazon S3 Storage Inventory in the Amazon Simple Storage Service Developer Guide. | November 29, 2016 |
| Amazon S3 Analytics – Storage Class Analysis | The new Amazon S3 analytics – storage class analysis feature observes data access patterns to help you determine when to transition less frequently accessed STANDARD storage to the STANDARD_IA (IA, for infrequent access) storage class. After storage class analysis observes the infrequent access patterns of a filtered set of data over a period of time, you can use the analysis results to help you improve your lifecycle policies. This feature also includes a detailed daily analysis of your storage usage at the specified bucket, prefix, or tag level that you can export to a S3 bucket. The following new API operations are for storage class analysis:  
  - `DeleteBucketAnalyticsConfiguration` (p. 45)  
  - `GetBucketAnalyticsConfiguration` (p. 93)  
  - `PutBucketAnalyticsConfiguration` (p. 267)  
  - `ListBucketAnalyticsConfigurations` (p. 196)  
  For more information, see Amazon S3 Analytics – Storage Class Analysis in the Amazon Simple Storage Service Developer Guide. | November 29, 2016 |
<p>| Added S3 Glacier retrieval options to RestoreObject (p. 378) | Amazon S3 now supports Expedited and Bulk data retrievals in addition to Standard retrievals when restoring objects archived to S3 Glacier. For more information, see Restoring Archived Objects in the Amazon Simple Storage Service Developer Guide. | November 21, 2016 |
| US East (Ohio) Region | Amazon S3 is now available in the US East (Ohio) Region. For more information about Amazon S3 regions and endpoints, see Regions and Endpoints in the AWS General Reference. | October 17, 2016 |</p>
<table>
<thead>
<tr>
<th>Change</th>
<th>Description</th>
<th>Release Date</th>
</tr>
</thead>
<tbody>
<tr>
<td>Asia Pacific (Mumbai) region</td>
<td>Amazon S3 is now available in the Asia Pacific (Mumbai) region. For more information about Amazon S3 regions and endpoints, see Regions and Endpoints in the AWS General Reference.</td>
<td>June 27, 2016</td>
</tr>
<tr>
<td>GET Bucket (List Objects) API revised</td>
<td>The GET Bucket (List Objects) API has been revised. We recommend that you use the new version, GET Bucket (List Objects) version 2. For more information, see ListObjectsV2 (p. 230).</td>
<td>May 4, 2016</td>
</tr>
<tr>
<td>Amazon S3 Transfer Acceleration</td>
<td>Amazon S3 Transfer Acceleration enables fast, easy, and secure transfers of files over long distances between your client and an S3 bucket. Transfer Acceleration takes advantage of Amazon CloudFront's globally distributed edge locations. For more information, see Transfer Acceleration in the Amazon Simple Storage Service Developer Guide. The following new API operations support Transfer Acceleration: GetBucketAccelerateConfiguration (p. 87) and PutBucketAccelerateConfiguration (p. 258).</td>
<td>April 19, 2016</td>
</tr>
<tr>
<td>Lifecycle support to remove expired object delete marker</td>
<td>Lifecycle configuration expiration action now allows you to direct Amazon S3 to remove expired object delete markers in versioned bucket. For more information, see Elements to Describe Lifecycle Actions in the Amazon Simple Storage Service Developer Guide.</td>
<td>March 16, 2016</td>
</tr>
<tr>
<td>Change</td>
<td>Description</td>
<td>Release Date</td>
</tr>
<tr>
<td>--------</td>
<td>-------------</td>
<td>--------------</td>
</tr>
</tbody>
</table>
| Bucket lifecycle configuration now supports the action to cancel incomplete multipart uploads | Bucket lifecycle configuration now supports the `AbortIncompleteMultipartUpload` action that you can use to direct Amazon S3 to cancel multipart uploads that don't complete within a specified number of days after being initiated. When a multipart upload becomes eligible for an abort operation, Amazon S3 deletes any uploaded parts and cancels the multipart upload. The following API operations have been updated to support the new action:  
• PutBucketLifecycleConfiguration (p. 292) – The XML configuration now allows you to specify the `AbortIncompleteMultipartUpload` action in a lifecycle configuration rule.  
• ListParts (p. 252) and CreateMultipartUpload (p. 33) – Both of these API operations now return two additional response headers (x-amz-abort-date, and x-amz-abort-rule-id) if the bucket has a lifecycle rule that specifies the `AbortIncompleteMultipartUpload` action. These headers in the response indicate when the initiated multipart upload will become eligible for an abort operation and which lifecycle rule is applicable.  
For conceptual information, see the following topics in the Amazon Simple Storage Service Developer Guide:  
• Aborting Incomplete Multipart Uploads Using a Bucket Lifecycle Policy  
• Elements to Describe Lifecycle Actions | March 16, 2016 |
<p>| Amazon S3 Signature Version 4 now supports unsigned payloads | Amazon S3 Signature Version 4 now supports unsigned payloads when authenticating requests using the Authorization header. Because you don't sign the payload, it does not provide the same security that comes with payload signing, but it provides similar performance characteristics as signature version 2. For more information, see Signature Calculations for the Authorization Header: Transferring Payload in a Single Chunk (AWS Signature Version 4) (p. 737). | January 15, 2016 |
| Asia Pacific (Seoul) region | Amazon S3 is now available in the Asia Pacific (Seoul) region. For more information about Amazon S3 regions and endpoints, see Regions and Endpoints in the AWS General Reference. | January 6, 2016 |
| Renamed the US Standard region | Changed the region name string from US Standard to US East (N. Virginia). This is only a region name update, there is no change in the functionality. | December 11, 2015 |</p>
<table>
<thead>
<tr>
<th>Change</th>
<th>Description</th>
<th>Release Date</th>
</tr>
</thead>
<tbody>
<tr>
<td>New storage class</td>
<td>Amazon S3 now offers a new storage class, STANDARD_IA (IA, for infrequent access) for storing objects. This storage class is optimized for long-lived and less frequently accessed data. For more information, see Storage Classes in the Amazon Simple Storage Service Developer Guide. Lifecycle configuration feature updates now allow you to transition objects to the STANDARD_IA storage class. For more information, see Object Lifecycle Management in the Amazon Simple Storage Service Developer Guide. Previously, the cross-region replication feature used the storage class of the source object for object replicas. Now, when you configure cross-region replication you can specify a storage class for the object replica created in the destination bucket. For more information, see Cross-Region Replication in the Amazon Simple Storage Service Developer Guide.</td>
<td>September 16, 2015</td>
</tr>
<tr>
<td>Event notifications</td>
<td>Amazon S3 event notifications have been updated to add notifications when objects are deleted and to add filtering on object names with prefix and suffix matching. For the relevant API operations, see PutBucketNotificationConfiguration (p. 308), and GetBucketNotificationConfiguration (p. 128). For more information, see Configuring Amazon S3 Event Notifications in the Amazon Simple Storage Service Developer Guide.</td>
<td>July 28, 2015</td>
</tr>
<tr>
<td>Cross-region replication</td>
<td>Amazon S3 now supports cross-region replication. Cross-region replication is the automatic, asynchronous copying of objects across buckets in different AWS regions. For the relevant API operations, see PutBucketReplication (p. 320), GetBucketReplication (p. 139) and DeleteBucketReplication (p. 63). For more information, see Enabling Cross-Region Replication in the Amazon Simple Storage Service Developer Guide.</td>
<td>March 24, 2015</td>
</tr>
<tr>
<td>Event notifications</td>
<td>Amazon S3 now supports new event types and destinations in a bucket notification configuration. Prior to this release, Amazon S3 supported only the s3:ReducedRedundancyLostObject event type and an Amazon SNS topic as the destination. For more information about the new event types, go to Setting Up Notification of Bucket Events in the Amazon Simple Storage Service Developer Guide. For the relevant API operations, see PutBucketNotificationConfiguration (p. 308) and GetBucketNotificationConfiguration (p. 128).</td>
<td>November 13, 2014</td>
</tr>
<tr>
<td>Change</td>
<td>Description</td>
<td>Release Date</td>
</tr>
<tr>
<td>---------------------------------------------</td>
<td>------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
<td>--------------------</td>
</tr>
</tbody>
</table>
| Server-side encryption with AWS Key Management Service (KMS) | Amazon S3 now supports server-side encryption using AWS Key Management Service (KMS). With server-side encryption with KMS, you manage the envelope key through KMS, and Amazon S3 calls KMS to access the envelope key within the permissions you set.  

For more information about server-side encryption with KMS, see Protecting Data Using Server-Side Encryption with AWS Key Management Service in the Amazon Simple Storage Service Developer Guide.  

The following Amazon S3 REST API operations support headers related to KMS.  
• PutObject (p. 343)  
• CopyObject (p. 17)  
• POST Object (p. 769)  
• CreateMultipartUpload (p. 33)  
• UploadPart (p. 395) | November 12, 2014 |
| Europe (Frankfurt) Region | Amazon S3 is now available in the Europe (Frankfurt) Region. | October 23, 2014 |
| Server-side encryption with customer-provided encryption keys | Amazon S3 now supports server-side encryption using customer-provided encryption keys (SSE-C). Server-side encryption enables you to request Amazon S3 to encrypt your data at rest. When using SSE-C, Amazon S3 encrypts your objects with the custom encryption keys that you provide. Since Amazon S3 performs the encryption for you, you get the benefits of using your own encryption keys without the cost of writing or executing your own encryption code.  

For more information about SSE-C, go to Server-Side Encryption (Using Customer-Provided Encryption Keys) in the Amazon Simple Storage Service Developer Guide.  

The following Amazon S3 REST API operations support headers related to SSE-C.  
• GetObject (p. 154)  
• HeadObject (p. 187)  
• PutObject (p. 343)  
• CopyObject (p. 17)  
• POST Object (p. 769)  
• CreateMultipartUpload (p. 33)  
• UploadPart (p. 395)  
• UploadPartCopy (p. 401) | June 12, 2014 |
<table>
<thead>
<tr>
<th>Change</th>
<th>Description</th>
<th>Release Date</th>
</tr>
</thead>
</table>
| Lifecycle support for versioning           | Prior to this release lifecycle configuration was supported only on nonversioned buckets. Now you can configure lifecycle on both the nonversioned and versioning-enabled buckets.  

For more information, go to Object Lifecycle Management in the Amazon Simple Storage Service Developer Guide.  

The related API operations, see PutBucketLifecycleConfiguration (p. 292), GetBucketLifecycleConfiguration (p. 113), and DeleteBucketLifecycle (p. 55). | May 20, 2014 |
| Amazon S3 now supports Signature Version 4 | Amazon S3 now supports Signature Version 4 (SigV4) in all regions, the latest specification for how to sign and authenticate AWS requests.  

For more information, see Authenticating Requests (AWS Signature Version 4) (p. 733). | January 30, 2014 |
| Amazon S3 list actions now support encoding-type request parameter | The following Amazon S3 list actions now support encoding-type optional request parameter.  

ListObjects (p. 222)  

ListObjectVersions (p. 240)  

ListMultipartUploads (p. 214)  

ListParts (p. 252)  

An object key can contain any Unicode character; however, the XML 1.0 parser cannot parse some characters, such as characters with an ASCII value from 0 to 10. For characters that are not supported in XML 1.0, you can add this parameter to request that Amazon S3 encode the keys in the response. | November 1, 2013 |
<p>| SOAP Support Over HTTP Deprecated         | SOAP support over HTTP is deprecated, but it is still available over HTTPS. New Amazon S3 features will not be supported for SOAP. We recommend that you use either the REST API or the AWS SDKs. | September 19, 2013 |</p>
<table>
<thead>
<tr>
<th>Change</th>
<th>Description</th>
<th>Release Date</th>
</tr>
</thead>
<tbody>
<tr>
<td>Root domain support</td>
<td>Amazon S3 now supports hosting static websites at the root domain. Visitors to your website can access your site from their browser without specifying “www” in the web address (e.g., “example.com”). Many customers already host static websites on Amazon S3 that are accessible from a “www” subdomain (e.g., “www.example.com”). Previously, to support root domain access, you needed to run your own web server to proxy root domain requests from browsers to your website on Amazon S3. Running a web server to proxy requests introduces additional costs, operational burden, and another potential point of failure. Now, you can take advantage of the high availability and durability of Amazon S3 for both “www” and root domain addresses.</td>
<td>December 27, 2012</td>
</tr>
<tr>
<td>Support for Archiving Data</td>
<td>Amazon S3 now supports a storage option that enables you to utilize Amazon Glacier’s low-cost storage service for data archival. To archive objects, you define archival rules identifying objects and a timeline when you want Amazon S3 to archive these objects to S3 Glacier. You can easily set the rules on a bucket using the Amazon S3 console or programmatically using the Amazon S3 API or AWS SDKs.</td>
<td>November 13, 2012</td>
</tr>
<tr>
<td>for Amazon Glacier</td>
<td>To support data archival rules, Amazon S3 lifecycle management API has been updated. For more information, see PutBucketLifecycleConfiguration (p. 292). After you archive objects, you must first restore a copy before you can access the data. Amazon S3 offers a new API for you to initiate a restore. For more information, see RestoreObject (p. 378). For conceptual information, go to Object Lifecycle Management in the Amazon Simple Storage Service Developer Guide.</td>
<td></td>
</tr>
<tr>
<td>Support for Website Page</td>
<td>For a bucket that is configured as a website, Amazon S3 now supports redirecting a request for an object to another object in the same bucket or to an external URL. You can configure redirect by adding the x-amz-website-redirect-location metadata to the object.</td>
<td>October 4, 2012</td>
</tr>
<tr>
<td>Redirects</td>
<td>The object upload API operations PutObject (p. 343), CreateMultipartUpload (p. 33), and POST Object (p. 769) allow you to configure the x-amz-website-redirect-location object metadata.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>For conceptual information, go to How to Configure Website Page Redirects in the Amazon Simple Storage Service Developer Guide.</td>
<td></td>
</tr>
<tr>
<td>Change</td>
<td>Description</td>
<td>Release Date</td>
</tr>
<tr>
<td>--------</td>
<td>-------------</td>
<td>--------------</td>
</tr>
<tr>
<td>Cross-Origin Resource Sharing (CORS) support</td>
<td>Amazon S3 now supports Cross-Origin Resource Sharing (CORS). CORS defines a way in which client web applications that are loaded in one domain can interact with or access resources in a different domain. With CORS support in Amazon S3, you can build rich client-side web applications on top of Amazon S3 and selectively allow cross-domain access to your Amazon S3 resources. For more information, see <a href="https://docs.aws.amazon.com/AmazonS3/latest/dev/CORS.html">Enabling Cross-Origin Resource Sharing</a> in the Amazon Simple Storage Service Developer Guide.</td>
<td>August 31, 2012</td>
</tr>
<tr>
<td>Cost Allocation Tagging support</td>
<td>Amazon S3 now supports cost allocation tagging, which allows you to label S3 buckets so you can more easily track their cost against projects or other criteria. For more information, see <a href="https://docs.aws.amazon.com/AmazonS3/latest/dev/Tagging.html">Cost Allocation Tagging</a> in the Amazon Simple Storage Service Developer Guide.</td>
<td>August 21, 2012</td>
</tr>
<tr>
<td>Object Expiration support</td>
<td>You can use Object Expiration to schedule automatic removal of data after a configured time period. You set object expiration by adding lifecycle configuration to a bucket. For more information, see <a href="https://docs.aws.amazon.com/AmazonS3/latest/dev/LifeCycle.html">Transitioning Objects: General Considerations</a> in the Amazon Simple Storage Service Developer Guide.</td>
<td>December 27, 2011</td>
</tr>
<tr>
<td>New Region supported</td>
<td>Amazon S3 now supports the South America (São Paulo) region. For more information, see <a href="https://docs.aws.amazon.com/AmazonS3/latest/dev/UsingWebsiteInterfaces.html">Buckets and Regions</a> in the Amazon Simple Storage Service Developer Guide.</td>
<td>December 14, 2011</td>
</tr>
<tr>
<td>Multi-Object Delete</td>
<td>Amazon S3 now supports Multi-Object Delete API that enables you to delete multiple objects in a single request. With this feature, you can remove large numbers of objects from Amazon S3 more quickly than using multiple individual DELETE requests. For more information about the API see, see <a href="https://docs.aws.amazon.com/AmazonS3/latest/API/ApiExampleDeleteObjects.html">DeleteObjects</a>. For conceptual information about the delete operation, see <a href="https://docs.aws.amazon.com/AmazonS3/latest/dev/LifeCycle.html">Deleting Objects</a> in the Amazon Simple Storage Service Developer Guide.</td>
<td>December 7, 2011</td>
</tr>
<tr>
<td>New region supported</td>
<td>Amazon S3 now supports the US West (Oregon) region. For more information, see <a href="https://docs.aws.amazon.com/AmazonS3/latest/dev/UsingWebsiteInterfaces.html">Buckets and Regions</a> in the Amazon Simple Storage Service Developer Guide.</td>
<td>November 8, 2011</td>
</tr>
<tr>
<td>Server-side encryption support</td>
<td>Amazon S3 now supports server-side encryption. It enables you to request Amazon S3 to encrypt your data at rest, that is, encrypt your object data when Amazon S3 writes your data to disks in its data centers. To request server-side encryption, you must add the <code>x-amz-server-side-encryption</code> header to your request. To learn more about data encryption, go to <a href="https://docs.aws.amazon.com/AmazonS3/latest/dev/DataEncryption.html">Using Data Encryption</a> in the Amazon Simple Storage Service Developer Guide.</td>
<td>October 17, 2011</td>
</tr>
<tr>
<td>Change</td>
<td>Description</td>
<td>Release Date</td>
</tr>
<tr>
<td>--------</td>
<td>-------------</td>
<td>--------------</td>
</tr>
<tr>
<td>Multipart Upload API extended to enable copying objects up to 5 TB</td>
<td>Prior to this release, Amazon S3 API supported copying objects (see CopyObject (p. 17)) of up to 5 GB in size. To enable copying objects larger than 5 GB, Amazon S3 extends the multipart upload API with a new operation, Upload Part (Copy). You can use this multipart upload operation to copy objects up to 5 TB in size. For conceptual information about multipart upload, go to Uploading Objects Using Multipart Upload in the Amazon Simple Storage Service Developer Guide. To learn more about the new API, see UploadPartCopy (p. 401).</td>
<td>June 21, 2011</td>
</tr>
<tr>
<td>SOAP API calls over HTTP disabled</td>
<td>To increase security, SOAP API calls over HTTP are disabled. Authenticated and anonymous SOAP requests must be sent to Amazon S3 using SSL.</td>
<td>June 6, 2011</td>
</tr>
</tbody>
</table>
| Support for hosting static websites in Amazon S3 | Amazon S3 introduces enhanced support for hosting static websites. This includes support for index documents and custom error documents. When using these features, requests to the root of your bucket or a subfolder (e.g., http://mywebsite.com/subfolder) returns your index document instead of the list of objects in your bucket. If an error is encountered, Amazon S3 returns your custom error message instead of an Amazon S3 error message. For API information to configure your bucket as a website, see the following sections:  
  - PutBucketWebsite (p. 336)  
  - GetBucketWebsite (p. 151)  
  - DeleteBucketWebsite (p. 67)  
For conceptual overview, go to Hosting Websites on Amazon S3 in the Amazon Simple Storage Service Developer Guide. | February 17, 2011 |
<p>| Response Header API Support | The GET Object REST API now allows you to change the response headers of the REST GET Object request for each request. That is, you can alter object metadata in the response, without altering the object itself. For more information, see GetObject (p. 154). | January 14, 2011 |
| Large Object Support | Amazon S3 has increased the maximum size of an object you can store in an S3 bucket from 5 GB to 5 TB. If you are using the REST API you can upload objects of up to 5 GB size in a single PUT operation. For larger objects, you must use the Multpart Upload REST API to upload objects in parts. For conceptual information, go to Uploading Objects Using Multipart Upload in the Amazon Simple Storage Service Developer Guide. For multipart upload API information, see CreateMultipartUpload (p. 33), UploadPart (p. 395), CompleteMultipartUpload (p. 11), ListParts (p. 252), and ListMultipartUploads (p. 214) | December 9, 2010 |</p>
<table>
<thead>
<tr>
<th>Change</th>
<th>Description</th>
<th>Release Date</th>
</tr>
</thead>
<tbody>
<tr>
<td>Multipart upload</td>
<td>Multipart upload enables faster, more flexible uploads into Amazon S3. It allows you to upload a single object as a set of parts. For conceptual information, go to Uploading Objects Using Multipart Upload in the Amazon Simple Storage Service Developer Guide. For multipart upload API information, see CreateMultipartUpload (p. 33), UploadPart (p. 395), CompleteMultipartUpload (p. 11), ListParts (p. 252), and ListMultipartUploads (p. 214)</td>
<td>November 10, 2010</td>
</tr>
<tr>
<td>Notifications</td>
<td>The Amazon S3 notifications feature enables you to configure a bucket so that Amazon S3 publishes a message to an Amazon Simple Notification Service (SNS) topic when Amazon S3 detects a key event on a bucket. For more information, see GET Bucket notification (p. 128) and PUT Bucket notification (p. 128).</td>
<td>July 14, 2010</td>
</tr>
<tr>
<td>Bucket policies</td>
<td>Bucket policies is an access management system you use to set access permissions on buckets, objects, and sets of objects. This functionality supplements and in many cases replaces access control lists.</td>
<td>July 6, 2010</td>
</tr>
<tr>
<td>Reduced Redundancy</td>
<td>Amazon S3 now enables you to reduce your storage costs by storing objects in Amazon S3 with reduced redundancy. For more information, see PUT Object (p. 343).</td>
<td>May 12, 2010</td>
</tr>
<tr>
<td>New region supported</td>
<td>Amazon S3 now supports the Asia Pacific (Singapore) region and therefore new location constraints. For more information, see GET Bucket location (p. 117) and PUT Bucket (p. 28).</td>
<td>April 28, 2010</td>
</tr>
<tr>
<td>Object Versioning</td>
<td>This release introduces object Versioning. All objects now have a key and a version. If you enable versioning for a bucket, Amazon S3 gives all objects added to a bucket a unique version ID. This feature enables you to recover from unintended overwrites and deletions. For more information, see GET Object (p. 154), DELETE Object (p. 69), PUT Object (p. 343), PUT Object Copy (p. 17), or POST Object (p. 769). The SOAP API does not support versioned objects.</td>
<td>February 8, 2010</td>
</tr>
<tr>
<td>New region supported</td>
<td>Amazon S3 now supports the US-West (Northern California) region. The new endpoint is s3-us-west-1.amazonaws.com. For more information, see How to Select a Region for Your Buckets in the Amazon Simple Storage Service Developer Guide.</td>
<td>December 2, 2009</td>
</tr>
<tr>
<td>C# Library Support</td>
<td>AWS now provides Amazon S3 C# libraries, sample code, tutorials, and other resources for software developers who prefer to build applications using language-specific API operations instead of REST or SOAP. These libraries provide basic functions (not included in the REST or SOAP APIs), such as request authentication, request retries, and error handling so that it’s easier to get started.</td>
<td>November 11, 2009</td>
</tr>
</tbody>
</table>
Technical documents reorganized

The API reference has been split out of the *Amazon S3 Developer Guide*. Now, on the documentation landing page, *Amazon Simple Storage Service Documentation*, you can select the document you want to view. When viewing the documents online, the links in one document will take you, when appropriate, to one of the other guides.

<table>
<thead>
<tr>
<th>Change</th>
<th>Description</th>
<th>Release Date</th>
</tr>
</thead>
<tbody>
<tr>
<td>Technical documents reorganized</td>
<td>The API reference has been split out of the <em>Amazon S3 Developer Guide</em>. Now, on the documentation landing page, <em>Amazon Simple Storage Service Documentation</em>, you can select the document you want to view. When viewing the documents online, the links in one document will take you, when appropriate, to one of the other guides.</td>
<td>September 16, 2009</td>
</tr>
</tbody>
</table>
Appendix

Topics

- Appendix: SelectObjectContent Response (p. 854)
- Appendix: OPTIONS object (p. 864)
- Appendix: SOAP API (p. 866)
- Appendix: Lifecycle Configuration APIs (Deprecated) (p. 893)
Appendix: SelectObjectContent Response

Description

The Amazon S3 Select operation filters the contents of an Amazon S3 object based on a simple structured query language (SQL) statement. Given the response size of this operation is unknown, Amazon S3 Select streams the response as a series of messages and includes a Transfer-Encoding header with chunked as its value in the response.

For more information about Amazon S3 Select, see Selecting Content from Objects in the Amazon Simple Storage Service Developer Guide.

For more information about using SQL with Amazon S3 Select, see SQL Reference for Amazon S3 Select and S3 Glacier Select in the Amazon Simple Storage Service Developer Guide.

Responses

A successful Amazon S3 Select Operation returns 200 OK status code.

Response Headers

This implementation of the operation uses only response headers that are common to most responses. For more information, see Common Response Headers (p. 814).

Response Body

Since the Amazon S3 Select response size is unknown, Amazon S3 streams the response as a series of messages and includes a Transfer-Encoding header with chunked as its value in the response. The following example shows the response format at the top level:

```xml
<Message 1>
<Message 2>
<Message 3>
......
<Message n>
```

Each message consists of two sections: the prelude and the data. The prelude section consists of 1) the total byte-length of the message, and 2) the combined byte-length of all the headers. The data section consists of 1) the headers, and 2) a payload.

Each section ends with a 4-byte big-endian integer checksum (CRC). Amazon S3 Select uses CRC32 (often referred to as GZIP CRC32) to calculate both CRCs. For more information about CRC32, see GZIP file format specification version 4.3.

Total message overhead including the prelude and both checksums is 16 bytes.

**Note**

All integer values within messages are in network byte order, or big-endian order.

The following diagram shows the components that make up a message and a header. Note that there are multiple headers per message.
Note
For Amazon S3 Select, the header value type is always 7 (type=String). For this type, the header value consists of two components, a 2-byte big-endian integer length, and a UTF-8 string that is of that byte-length. The following diagram shows the components that make up Amazon S3 Select headers.

Amazon S3 Select Headers (type=String):

Payload byte-length calculations (these two calculations are equivalent):

- payload_length = total_length - header_length - sizeOf(total_length) - sizeOf(header_length) - sizeOf(prelude_crc) - sizeOf(message_crc)
- payload_length = total_length - header_length - 16

Each message contains the following components:

- **Prelude**: Always fixed size of 8 bytes (two fields of 4 bytes each):
• **First four bytes**: Total byte-length: Big-endian integer byte-length of the entire message (including the 4-byte total length field itself).

• **Second four bytes**: Headers byte-length: Big-endian integer byte-length of the headers portion of the message (excluding the headers length field itself).

• **Prelude CRC**: 4-byte big-endian integer checksum (CRC) for the prelude portion of the message (excluding the CRC itself). The prelude has a separate CRC from the message CRC (see below), to ensure that corrupted byte-length information can be detected immediately, without causing pathological buffering behavior.

• **Headers**: A set of metadata annotating the message, such as the message type, payload format, and so on. Messages can have multiple headers, so this portion of the message can have different byte-lengths depending on the message type. Headers are key-value pairs, where both the key and value are UTF-8 strings. Headers can appear in any order within the headers portion of the message, and any given header type can only appear once.

For Amazon S3 Select, following is a list of header names and the set of valid values depending on the message type.

• **MessageType Header**:
  - **HeaderName** => `:message-type`
  - **Valid HeaderValues** => "error", "event"

• **EventType Header**:
  - **HeaderName** => `:event-type`
  - **Valid HeaderValues** => "Records", "Cont", "Progress", "Stats", "End"

• **ErrorCode Header**:
  - **HeaderName** => `:error-code`
  - **Valid HeaderValues** => Error Code from the table in the List of SELECT Object Content Error Codes (p. 825) section.

• **ErrorMessage Header**:
  - **HeaderName** => `:error-message`
  - **Valid HeaderValues** => Error message returned by the service, to help diagnose request-level errors.

• **Payload**: Can be anything.

• **Message CRC**: 4-byte big-endian integer checksum (CRC) from the start of the message to the start of the checksum (that is, everything in the message excluding the message CRC itself).

Each header contains the following components. There can be multiple headers per message.

• **Header Name Byte-Length**: Byte-length of the header name.

• **Header Name**: Name of the header, indicating the header type. Valid values: `:message-type` `:event-type` `:error-code` `:error-message`

• **Header Value Type**: Enum indicating the header value type. For Amazon S3 Select, this is always 7.

• **Value String Byte-Length**: (For Amazon S3 Select) Byte-length of the header value string.

• **Header Value String**: (For Amazon S3 Select) Value of the header string. Valid values for this field vary based on the type of the header. See the sections below for valid values for each header type and message type.

For Amazon S3 Select, responses can be messages of the following types:

• **Records message**: Can contain a single record, partial records, or multiple records. Depending on the size of the result, a response can contain one or more of these messages.
• **Continuation message**: Amazon S3 periodically sends this message to keep the TCP connection open. These messages appear in responses at random. The client must detect the message type and process accordingly.

• **Progress message**: Amazon S3 periodically sends this message, if requested. It contains information about the progress of a query that has started but has not yet completed.

• **Stats message**: Amazon S3 sends this message at the end of the request. It contains statistics about the query.

• **End message**: Indicates that the request is complete, and no more messages will be sent. You should not assume that the request is complete until the client receives an **End** message.

• **RequestLevelError message**: Amazon S3 sends this message if the request failed for any reason. It contains the error code and error message for the failure. If Amazon S3 sends a RequestLevelError message, it doesn’t send an **End** message.

The following sections explain the structure of each message type in more detail.

For sample code and unit tests that use this protocol, see [AWS C Event Stream](https://github.com/aws/aws-sdk-cpp) on the GitHub website.

**Records Message**

**Header specification**

Records messages contain three headers, as follows:
Payload specification

Records message payloads can contain a single record, partial records, or multiple records.

Continuation Message

Header specification

Continuation messages contain two headers, as follows:
**Payload specification**

Continuation messages have no payload.

**Progress Message**

**Header specification**

Progress messages contain three headers, as follows:

<table>
<thead>
<tr>
<th>Event-Type</th>
<th>Message-Type</th>
<th>2006-03-01 API Version</th>
<th>859</th>
</tr>
</thead>
<tbody>
<tr>
<td>11</td>
<td>&quot;event-type&quot;</td>
<td>7</td>
<td>4</td>
</tr>
<tr>
<td>1 byte</td>
<td>11 bytes</td>
<td>1 byte</td>
<td>2 bytes</td>
</tr>
<tr>
<td>13</td>
<td>&quot;message-type&quot;</td>
<td>7</td>
<td>5</td>
</tr>
<tr>
<td>1 byte</td>
<td>13 bytes</td>
<td>1 byte</td>
<td>2 bytes</td>
</tr>
</tbody>
</table>

Progress message payload is an XML document containing information about the progress of a request.

- **BytesScanned** => Number of bytes that have been processed before being uncompressed (if the file is compressed).
- **BytesProcessed** => Number of bytes that have been processed after being uncompressed (if the file is compressed).
- **BytesReturned** => Current number of bytes of records payload data returned by Amazon S3.

For uncompressed files, **BytesScanned** and **BytesProcessed** are equal.

Example:

```xml
<?xml version="1.0" encoding="UTF-8"?>
<Progress>
  <BytesScanned>512</BytesScanned>
  <BytesProcessed>1024</BytesProcessed>
  <BytesReturned>1024</BytesReturned>
</Progress>
```
Stats Message

Header specification

Stats messages contain three headers, as follows:

Payload specification

Stats message payload is an XML document containing information about a request's stats when processing is complete.

- **BytesScanned** => Number of bytes that have been processed before being uncompressed (if the file is compressed).
- **BytesProcessed** => Number of bytes that have been processed after being uncompressed (if the file is compressed).
- **BytesReturned** => Total number of bytes of records payload data returned by Amazon S3.

For uncompressed files, **BytesScanned** and **BytesProcessed** are equal.

Example:

```xml
<?xml version="1.0" encoding="UTF-8"?>
```
End Message

Header specification

End messages contain two headers, as follows:

 Payload specification

End messages have no payload.

Request Level Error Message

Header specification

Request-level error messages contain three headers, as follows:
For a list of possible error codes and error messages, see the List of SELECT Object Content Error Codes (p. 825).

Payload specification

Request-level error messages have no payload.

## Related Resources

- the section called “SelectObjectContent” (p. 387)
- the section called “GetObject” (p. 154)
- the section called “GetBucketLifecycleConfiguration” (p. 113)
- the section called “PutBucketLifecycleConfiguration” (p. 292)
Appendix: OPTIONS object

Description

A browser can send this preflight request to Amazon S3 to determine if it can send an actual request with the specific origin, HTTP method, and headers.

Amazon S3 supports cross-origin resource sharing (CORS) by enabling you to add a cors subresource on a bucket. When a browser sends this preflight request, Amazon S3 responds by evaluating the rules that are defined in the cors configuration.

If cors is not enabled on the bucket, then Amazon S3 returns a 403 Forbidden response.

For more information about CORS, go to Enabling Cross-Origin Resource Sharing in the Amazon Simple Storage Service Developer Guide.

Requests

Syntax

OPTIONS /ObjectName HTTP/1.1
Host: BucketName.s3.amazonaws.com
Origin: Origin
Access-Control-Request-Method: HTTPMethod
Access-Control-Request-Headers: RequestHeader

Request Parameters

This operation does not introduce any specific request parameters, but it may contain any request parameters that are required by the actual request.

Request Headers

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
<th>Required</th>
</tr>
</thead>
<tbody>
<tr>
<td>Origin</td>
<td>Identifies the origin of the cross-origin request to Amazon S3.</td>
<td>Yes</td>
</tr>
<tr>
<td></td>
<td>For example, <a href="http://www.example.com">http://www.example.com</a>.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Type: String</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Default: None</td>
<td></td>
</tr>
<tr>
<td>Access-Control-Request-Method</td>
<td>Identifies what HTTP method will be used in the actual request.</td>
<td>Yes</td>
</tr>
<tr>
<td></td>
<td>Type: String</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Default: None</td>
<td></td>
</tr>
<tr>
<td>Access-Control-Request-Headers</td>
<td>A comma-delimited list of HTTP headers that will be sent in the actual request.</td>
<td>No</td>
</tr>
<tr>
<td></td>
<td>For example, to put an object with server-side encryption, this preflight request will determine if it can include the x-amz-server-side-encryption header with the request.</td>
<td></td>
</tr>
</tbody>
</table>
### Responses

#### Request Elements

This implementation of the operation does not use request elements.

#### Responses

##### Response Headers

<table>
<thead>
<tr>
<th>Header</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Access-Control-Allow-Origin</td>
<td>The origin you sent in your request. If the origin in your request is not allowed, Amazon S3 will not include this header in the response. Type: String</td>
</tr>
<tr>
<td>Access-Control-Max-Age</td>
<td>How long, in seconds, the results of the preflight request can be cached. Type: String</td>
</tr>
<tr>
<td>Access-Control-Allow-Methods</td>
<td>The HTTP method that was sent in the original request. If the method in the request is not allowed, Amazon S3 will not include this header in the response. Type: String</td>
</tr>
<tr>
<td>Access-Control-Allow-Headers</td>
<td>A comma-delimited list of HTTP headers that the browser can send in the actual request. If any of the requested headers is not allowed, Amazon S3 will not include that header in the response, nor will the response contain any of the headers with the Access-Control prefix. Type: String</td>
</tr>
<tr>
<td>Access-Control-Expose-Headers</td>
<td>A comma-delimited list of HTTP headers. This header provides the JavaScript client with access to these headers in the response to the actual request. Type: String</td>
</tr>
</tbody>
</table>

##### Response Elements

This implementation of the operation does not return response elements.
Examples

Example: Send a preflight OPTIONS request to a cors enabled bucket

A browser can send this preflight request to Amazon S3 to determine if it can send the actual PUT request from http://www.example.com origin to the Amazon S3 bucket named examplebucket.

Sample Request

```
OPTIONS /exampleobject HTTP/1.1
Host: examplebucket.s3.amazonaws.com
Origin: http://www.example.com
Access-Control-Request-Method: PUT
```

Sample Response

```
HTTP/1.1 200 OK
x-amz-id-2: 6SvaESv3VULYPbIk5LLl7lSPPtSnNdGmknIX1HfU7uS2m1DF6td6KWNjYMXX
x-amz-request-id: BDC4B83DF5096B8E
Date: Wed, 21 Aug 2012 23:09:55 GMT
Etag: "1f1aaf1f11111111111111c1a11d11"
Access-Control-Allow-Origin: http://www.example.com
Access-Control-Allow-Methods: PUT
Access-Control-Expose-Headers: x-amz-request-id
Content-Length: 0
Server: AmazonS3
```

Related Resources

- GetBucketCors (p. 97)
- DeleteBucketCors (p. 47)
- PutBucketCors (p. 271)

Appendix: SOAP API

Note

SOAP support over HTTP is deprecated, but it is still available over HTTPS. New Amazon S3 features will not be supported for SOAP. We recommend that you use either the REST API or the AWS SDKs.

This section describes the SOAP API with respect to service, bucket, and object operations. Note that SOAP requests, both authenticated and anonymous, must be sent to Amazon S3 using SSL. Amazon S3 returns an error when you send a SOAP request over HTTP.

The latest Amazon S3 WSDL is available at http://doc.s3.amazonaws.com/2006-03-01/AmazonS3.wsdl.

Topics

- Operations on the Service (SOAP API) (p. 867)
- Operations on Buckets (SOAP API) (p. 868)
- Operations on Objects (SOAP API) (p. 876)
Operations on the Service (SOAP API)

**Note**
SOAP support over HTTP is deprecated, but it is still available over HTTPS. New Amazon S3 features will not be supported for SOAP. We recommend that you use either the REST API or the AWS SDKs.

This section describes operations you can perform on the Amazon S3 service.

**Topics**
- ListAllMyBuckets (SOAP API) (p. 867)

### ListAllMyBuckets (SOAP API)

**Note**
SOAP support over HTTP is deprecated, but it is still available over HTTPS. New Amazon S3 features will not be supported for SOAP. We recommend that you use either the REST API or the AWS SDKs.

The ListAllMyBuckets operation returns a list of all buckets owned by the sender of the request.

**Example**

**Sample Request**

```xml
<ListAllMyBuckets xmlns="http://doc.s3.amazonaws.com/2006-03-01">
  <AWSAccessKeyId>AKIAIOSFODNN7EXAMPLE</AWSAccessKeyId>
  <Timestamp>2006-03-01T12:00:00.183Z</Timestamp>
  <Signature>Iuyz3d3P0aTou39dzbqaEXAMPLE=</Signature>
</ListAllMyBuckets>
```

**Sample Response**

```xml
<ListAllMyBucketsResult xmlns="http://s3.amazonaws.com/doc/2006-03-01">
  <Owner>
    <ID>bcaf1ffdb86f41161ca5f16fd081034f</ID>
    <DisplayName>webfile</DisplayName>
  </Owner>
  <Buckets>
    <Bucket>
      <Name>quotes/</Name>
      <CreationDate>2006-02-03T16:45:00Z</CreationDate>
    </Bucket>
    <Bucket>
      <Name>samples</Name>
      <CreationDate>2006-02-03T16:41:58.000Z</CreationDate>
    </Bucket>
  </Buckets>
</ListAllMyBucketsResult>
```

**Response Body**

- **Owner:**

  This provides information that Amazon S3 uses to represent your identity for purposes of authentication and access control. ID is a unique and permanent identifier for the developer who
made the request. DisplayName is a human-readable name representing the developer who made the request. It is not unique, and might change over time. We recommend that you match your DisplayName to your Forum name.

- **Name:**
  The name of a bucket. Note that if one of your buckets was recently deleted, the name of the deleted bucket might still be present in this list for a period of time.

- **CreationDate:**
  The time that the bucket was created.

### Access Control

You must authenticate with a valid AWS Access Key ID. Anonymous requests are never allowed to list buckets, and you can only list buckets for which you are the owner.

### Operations on Buckets (SOAP API)

**Note**

SOAP support over HTTP is deprecated, but it is still available over HTTPS. New Amazon S3 features will not be supported for SOAP. We recommend that you use either the REST API or the AWS SDKs.

This section describes operations you can perform on Amazon S3 buckets.

#### Topics

- CreateBucket (SOAP API) (p. 868)
- DeleteBucket (SOAP API) (p. 869)
- ListBucket (SOAP API) (p. 870)
- GetBucketAccessControlPolicy (SOAP API) (p. 873)
- SetBucketAccessControlPolicy (SOAP API) (p. 874)
- GetBucketLoggingStatus (SOAP API) (p. 875)
- SetBucketLoggingStatus (SOAP API) (p. 875)

### CreateBucket (SOAP API)

**Note**

SOAP support over HTTP is deprecated, but it is still available over HTTPS. New Amazon S3 features will not be supported for SOAP. We recommend that you use either the REST API or the AWS SDKs.

The CreateBucket operation creates a bucket. Not every string is an acceptable bucket name. For information on bucket naming restrictions, see Working with Amazon S3 Buckets.

**Note**

To determine whether a bucket name exists, use ListBucket and set MaxKeys to 0. A NoSuchBucket response indicates that the bucket is available, an AccessDenied response indicates that someone else owns the bucket, and a Success response indicates that you own the bucket or have permission to access it.

**Example Create a bucket named "quotes"**

**Sample Request**
CreateBucket (SOAP API)

**Sample Response**

```
  <CreateBucketResponse/>
</CreateBucketResponse>
```

**Elements**

- **Bucket**: The name of the bucket you are trying to create.
- **AccessControlList**: The access control list for the new bucket. This element is optional. If not provided, the bucket is created with an access policy that give the requester FULL_CONTROL access.

**Access Control**

You must authenticate with a valid AWS Access Key ID. Anonymous requests are never allowed to create buckets.

**Related Resources**

- [ListBucket (SOAP API)](p. 870)

DeleteBucket (SOAP API)

**Note**

SOAP support over HTTP is deprecated, but it is still available over HTTPS. New Amazon S3 features will not be supported for SOAP. We recommend that you use either the REST API or the AWS SDKs.

The `DeleteBucket` operation deletes a bucket. All objects in the bucket must be deleted before the bucket itself can be deleted.

**Example**

This example deletes the "quotes" bucket.

**Sample Request**

```
<DeleteBucket xmlns="http://s3.amazonaws.com/doc/2006-03-01">
  <Bucket>quotes</Bucket>
  <AWSAccessKeyId>AKIAIOSFODNN7EXAMPLE</AWSAccessKeyId>
  <Timestamp>2006-03-01T12:00:00.183Z</Timestamp>
  <Signature>Iuyz3d3P0aTou39dzbqaEXAMPLE=</Signature>
</DeleteBucket>
```

**Sample Response**

API Version 2006-03-01

869
<!--DeleteBucketResponse xmlns="http://s3.amazonaws.com/doc/2006-03-01">
  <DeleteBucketResponse>
    <Code>204</Code>
    <Description>No Content</Description>
  </DeleteBucketResponse>
</DeleteBucketResponse>

Elements

- Bucket: The name of the bucket you want to delete.

Access Control

Only the owner of a bucket is allowed to delete it, regardless the access control policy on the bucket.

ListBucket (SOAP API)

**Note**

SOAP support over HTTP is deprecated, but it is still available over HTTPS. New Amazon S3 features will not be supported for SOAP. We recommend that you use either the REST API or the AWS SDKs.

The `ListBucket` operation returns information about some of the items in the bucket.

For a general introduction to the list operation, see the [Listing Object Keys](#).

Requests

This example lists up to 1000 keys in the "quotes" bucket that have the prefix "notes."

Syntax

```xml
<ListBucket xmlns="http://doc.s3.amazonaws.com/2006-03-01">
  <Bucket>quotes</Bucket>
  <Prefix>notes/</Prefix>
  <Delimiter>/</Delimiter>
  <MaxKeys>1000</MaxKeys>
  <AWSAccessKeyId>AKIAIOSFODNN7EXAMPLE</AWSAccessKeyId>
  <Timestamp>2006-03-01T12:00:00.183Z</Timestamp>
  <Signature>Iuyz3d3P0aTou39dzbqaEXAMPLE=</Signature>
</ListBucket>
```

Parameters

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
<th>Required</th>
</tr>
</thead>
<tbody>
<tr>
<td>prefix</td>
<td>Limits the response to keys which begin with the indicated prefix. You can use prefixes to separate a bucket into different sets of keys in a way similar to how a file system uses folders. Type: String Default: None</td>
<td>No</td>
</tr>
<tr>
<td>marker</td>
<td>Indicates where in the bucket to begin listing. The list will only include keys that occur lexicographically after marker. This is</td>
<td>No</td>
</tr>
<tr>
<td>Name</td>
<td>Description</td>
<td>Required</td>
</tr>
<tr>
<td>------------</td>
<td>-----------------------------------------------------------------------------</td>
<td>----------</td>
</tr>
<tr>
<td>next-key</td>
<td>convenient for pagination: To get the next page of results use the last key of the current page as the marker.</td>
<td></td>
</tr>
<tr>
<td>max-keys</td>
<td>The maximum number of keys you'd like to see in the response body. The server might return fewer than this many keys, but will not return more.</td>
<td>No</td>
</tr>
<tr>
<td>delimiter</td>
<td>Causes keys that contain the same string between the prefix and the first occurrence of the delimiter to be rolled up into a single result element in the CommonPrefixes collection. These rolled-up keys are not returned elsewhere in the response.</td>
<td>No</td>
</tr>
</tbody>
</table>

**Success Response**

This response assumes the bucket contains the following keys:

- notes/todos.txt
- notes/2005-05-23/customer_mtg_notes.txt
- notes/2005-05-23/phone_notes.txt
- notes/2005-05-28/sales_notes.txt

**Syntax**

```xml
<?xml version="1.0" encoding="UTF-8"?>
  <Name>backups</Name>
  <Prefix>notes/</Prefix>
  <MaxKeys>1000</MaxKeys>
  <Delimiter>/</Delimiter>
  <IsTruncated>false</IsTruncated>
  <Contents>
    <Key>notes/todos.txt</Key>
    <LastModified>2006-01-01T12:00:00.000Z</LastModified>
    <ETag>&quot;828ef3fda96f00ad9f27c383fc9ac7f&quot;</ETag>
    <Size>5126</Size>
    <StorageClass>STANDARD</StorageClass>
    <Owner>
      <ID>75aa57f09aa0c89e964f88c324e99d10f8e7faeef76c078efc7c6caea54ba06a</ID>
      <DisplayName>webfile</DisplayName>
    </Owner>
    <StorageClass>STANDARD</StorageClass>
  </Contents>
  <CommonPrefixes>
    <Prefix>notes/2005-05-23/</Prefix>
  </CommonPrefixes>
</ListBucketResult>
```
As you can see, many of the fields in the response echo the request parameters. IsTruncated, Contents, and CommonPrefixes are the only response elements that can contain new information.

### Response Elements

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Contents</td>
<td>Metadata about each object returned.</td>
</tr>
<tr>
<td></td>
<td>Type: XML metadata</td>
</tr>
<tr>
<td></td>
<td>Ancestor: ListBucketResult</td>
</tr>
<tr>
<td>CommonPrefixes</td>
<td>A response can contain CommonPrefixes only if you specify a delimiter. When you do, CommonPrefixes contains all (if there are any) keys between</td>
</tr>
<tr>
<td></td>
<td>Prefix and the next occurrence of the string specified by delimiter. In effect, CommonPrefixes lists keys that act like subdirectories in the</td>
</tr>
</tbody>
</table>
|              | directory specified by Prefix. For example, if prefix is notes/ and delimiter is a slash (/), in notes/summer/july, the common prefix is notes/summer/.
<p>|              | Type: String                                                                                                                                |
|              | Ancestor: ListBucketResult                                                                                                                   |
| Delimiter    | Causes keys that contain the same string between the prefix and the first occurrence of the delimiter to be rolled up into a single result element |
|              | in the CommonPrefixes collection. These rolled-up keys are not returned elsewhere in the response.                                            |
|              | Type: String                                                                                                                                |
|              | Ancestor: ListBucketResult                                                                                                                   |
| IsTruncated  | Specifies whether (true) or not (false) all of the results were returned. All of the results may not be returned if the number of results    |
|              | exceeds that specified by MaxKeys.                                                                                                           |
|              | Type: String                                                                                                                                |
|              | Ancestor: boolean                                                                                                                           |
| Marker       | Indicates where in the bucket to begin listing.                                                                                              |
|              | Type: String                                                                                                                                |
|              | Ancestor: ListBucketResult                                                                                                                   |
| MaxKeys      | The maximum number of keys returned in the response body.                                                                                      |
|              | Type: String                                                                                                                                |
|              | Ancestor: ListBucketResult                                                                                                                   |
| Name         | Name of the bucket.                                                                                                                          |
|              | Type: String                                                                                                                                |</p>
<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Ancestor: ListBucketResult</td>
<td>Ancestor: ListBucketResult</td>
</tr>
<tr>
<td>Prefix</td>
<td>Keys that begin with the indicated prefix.</td>
</tr>
<tr>
<td></td>
<td>Type: String</td>
</tr>
<tr>
<td></td>
<td>Ancestor: ListBucketResult</td>
</tr>
</tbody>
</table>

**Response Body**

For information about the list response, see [Listing Keys Response](#).

**Access Control**

To list the keys of a bucket you need to have been granted **READ** access on the bucket.

**GetBucketAccessControlPolicy (SOAP API)**

**Note**

SOAP support over HTTP is deprecated, but it is still available over HTTPS. New Amazon S3 features will not be supported for SOAP. We recommend that you use either the REST API or the AWS SDKs.

The **GetBucketAccessControlPolicy** operation fetches the access control policy for a bucket.

**Example**

This example retrieves the access control policy for the "quotes" bucket.

**Sample Request**

```xml
  <Bucket>quotes</Bucket>
  <AWSAccessKeyId>AKIAIOSFODNN7EXAMPLE</AWSAccessKeyId>
  <Timestamp>2006-03-01T12:00:00.183Z</Timestamp>
  <Signature>Iuyz3d3P0aTou39dzbqaEXAMPLE=</Signature>
</GetBucketAccessControlPolicy>
```

**Sample Response**

```xml
<AccessControlPolicy>
  <Owner>
    <ID>a9a7b886d6fd2441bf9b1c61be666e9</ID>
    <DisplayName>chriscustomer</DisplayName>
  </Owner>
  <AccessControlList>
    <Grant>
      <Grantee xsi:type="CanonicalUser">
        <ID>a9a7b886d6fd2441bf9b1c61be666e9</ID>
        <DisplayName>chriscustomer</DisplayName>
      </Grantee>
      <Permission>FULL_CONTROL</Permission>
    </Grant>
    <Grant>
      <Grantee xsi:type="Group">
        <URI>http://acs.amazonaws.com/groups/global/AllUsers</URI>
      </Grantee>
      <Permission>FULL_CONTROL</Permission>
    </Grant>
  </AccessControlList>
</AccessControlPolicy>
```
Response Body

The response contains the access control policy for the bucket. For an explanation of this response, see SOAP Access Policy.

Access Control

You must have READ_ACP rights to the bucket in order to retrieve the access control policy for a bucket.

**SetBucketAccessControlPolicy (SOAP API)**

*Note*

SOAP support over HTTP is deprecated, but it is still available over HTTPS. New Amazon S3 features will not be supported for SOAP. We recommend that you use either the REST API or the AWS SDKs.

The `SetBucketAccessControlPolicy` operation sets the Access Control Policy for an existing bucket. If successful, the previous Access Control Policy for the bucket is entirely replaced with the specified Access Control Policy.

**Example**

Give the specified user (usually the owner) FULL_CONTROL access to the "quotes" bucket.

**Sample Request**

```xml
  <Bucket>quotes</Bucket>
  <AccessControlList>
    <Grant>
      <Grantee xsi:type="CanonicalUser">
        <ID>a9a7b863000e241bf9b1c61be666e9</ID>
        <DisplayName>chriscustomer</DisplayName>
      </Grantee>
      <Permission>FULL_CONTROL</Permission>
    </Grant>
  </AccessControlList>
  <AWSAccessKeyId>AKIAIOSFODNN7EXAMPLE</AWSAccessKeyId>
  <Timestamp>2006-03-01T12:00:00.183Z</Timestamp>
  <Signature>Iuyz3d3P0aTou39dzbqaEXAMPLE=</Signature>
</SetBucketAccessControlPolicy>
```

**Sample Response**

```xml
  <GetBucketAccessControlPolicyResponse>
    <Code>200</Code>
    <Description>OK</Description>
  </GetBucketAccessControlPolicyResponse>
</GetBucketAccessControlPolicyResponse>
```

Access Control

You must have WRITE_ACP rights to the bucket in order to set the access control policy for a bucket.
GetBucketLoggingStatus (SOAP API)

**Note**
SOAP support over HTTP is deprecated, but it is still available over HTTPS. New Amazon S3 features will not be supported for SOAP. We recommend that you use either the REST API or the AWS SDKs.

The GetBucketLoggingStatus retrieves the logging status for an existing bucket.

For a general introduction to this feature, see [Server Logs](#).

**Example**

**Sample Request**

```xml
<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/"
  <soap:Body>
      <Bucket>mybucket</Bucket>
      <AWSAccessKeyId>YOUR_AWS_ACCESS_KEY_ID</AWSAccessKeyId>
      <Timestamp>2006-03-01T12:00:00.183Z</Timestamp>
      <Signature>YOUR_SIGNATURE_HERE</Signature>
    </GetBucketLoggingStatus>
  </soap:Body>
</soap:Envelope>
```

**Sample Response**

```xml
<?xml version="1.0" encoding="utf-8"?>
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
  <soapenv:Body>
      <LoggingEnabled>
        <TargetBucket>mylogs</TargetBucket>
        <TargetPrefix>mybucket-access_log-</TargetPrefix>
      </LoggingEnabled>
    </GetBucketLoggingStatusResponse>
  </soapenv:Body>
</soapenv:Envelope>
```

**Access Control**

Only the owner of a bucket is permitted to invoke this operation.

SetBucketLoggingStatus (SOAP API)

**Note**
SOAP support over HTTP is deprecated, but it is still available over HTTPS. New Amazon S3 features will not be supported for SOAP. We recommend that you use either the REST API or the AWS SDKs.
The `SetBucketLoggingStatus` operation updates the logging status for an existing bucket.

For a general introduction to this feature, see Server Logs.

**Example**

This sample request enables server access logging for the 'mybucket' bucket, and configures the logs to be delivered to 'mylogs' under prefix 'access_log-

**Sample Request**

```xml
<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/
  <soap:Body>
    <SetBucketLoggingStatus xmlns="http://doc.s3.amazonaws.com/2006-03-01">
      <Bucket>myBucket</Bucket>
      <AWSAccessKeyId>YOUR_AWS_ACCESS_KEY_ID</AWSAccessKeyId>
      <Timestamp>2006-03-01T12:00:00.183Z</Timestamp>
      <Signature>YOUR_SIGNATURE_HERE</Signature>
      <BucketLoggingStatus>
        <LoggingEnabled>
          <TargetBucket>mylogs</TargetBucket>
          <TargetPrefix>mybucket-access_log-</TargetPrefix>
        </LoggingEnabled>
      </BucketLoggingStatus>
    </SetBucketLoggingStatus>
  </soap:Body>
</soap:Envelope>
```

**Sample Response**

```xml
<?xml version="1.0" encoding="utf-8"?>
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/
  <soapenv:Header>
  </soapenv:Header>
  <soapenv:Body>
  </soapenv:Body>
</soapenv:Envelope>
```

**Access Control**

Only the owner of a bucket is permitted to invoke this operation.

**Operations on Objects (SOAP API)**

**Note**

SOAP support over HTTP is deprecated, but it is still available over HTTPS. New Amazon S3 features will not be supported for SOAP. We recommend that you use either the REST API or the AWS SDKs.

This section describes operations you can perform on Amazon S3 objects.
The `PutObjectInline` operation adds an object to a bucket. The data for the object is provided in the body of the SOAP message.

If an object already exists in a bucket, the new object will overwrite it because Amazon S3 stores the last write request. However, Amazon S3 is a distributed system. If Amazon S3 receives multiple write requests for the same object nearly simultaneously, all of the objects might be stored, even though only one wins in the end. Amazon S3 does not provide object locking; if you need this, make sure to build it into your application layer.

To ensure an object is not corrupted over the network, you can calculate the MD5 of an object, PUT it to Amazon S3, and compare the returned Etag to the calculated MD5 value.

`PutObjectInline` is not suitable for use with large objects. The system limits this operation to working with objects 1MB or smaller. `PutObjectInline` will fail with the `InlineDataTooLargeError` status code if the Data parameter encodes an object larger than 1MB. To upload large objects, consider using the non-inline `PutObject` API, or the REST API instead.

**Example**

This example writes some text and metadata into the "Nelson" object in the "quotes" bucket, give a user (usually the owner) FULL_CONTROL access to the object, and make the object readable by anonymous parties.

**Sample Request**

```xml
<PutObjectInline xmlns="http://doc.s3.amazonaws.com/2006-03-01">
  <Bucket>quotes</Bucket>
  <Key>Nelson</Key>
  <Metadata>
    <Name>Content-Type</Name>
    <Value>text/plain</Value>
  </Metadata>
  <Metadata>
    <Name>family</Name>
    <Value>Muntz</Value>
  </Metadata>
  <Data>aGEtaGE=</Data>
  <ContentLength>5</ContentLength>
  <AccessControlList>
    <Owner>
      <DisplayName>Amazon S3</DisplayName>
      <ID>864503722305745267</ID>
    </Owner>
    <Access>
      <Grant READ>
        <Grantee>
          <DisplayName>everyone</DisplayName>
          <ID>864503722305745267</ID>
        </Grantee>
      </Grant>
    </Access>
  </AccessControlList>
</PutObjectInline>
```
<Grant>
  <Grantee xsi:type="CanonicalUser">
    <ID>a9a7b886d6fde241bf9b1c61be666e9</ID>
    <DisplayName>chriscustomer</DisplayName>
  </Grantee>
  <Permission>FULL_CONTROL</Permission>
</Grant>

<Grant>
  <Grantee xsi:type="Group">
    <URI>http://acs.amazonaws.com/groups/global/AllUsers</URI>
  </Grantee>
  <Permission>READ</Permission>
</Grant>
</AccessControlList>

<AWSAccessKeyId>AKIAIOSFODNN7EXAMPLE</AWSAccessKeyId>
<Timestamp>2006-03-01T12:00:00.183Z</Timestamp>
<Signature>Iuyz3d3P0aTou39dzbqaEXAMPLE=</Signature>
</PutObjectInline>

Sample Response

  <ETag>&quot;828ef3fdfa96f00ad9f27c383fc9ac7f&quot;</ETag>
  <LastModified>2006-01-01T12:00:00.000Z</LastModified>
</PutObjectInlineResponse>

Elements

- **Bucket**: The bucket in which to add the object.
- **Key**: The key to assign to the object.
- **Metadata**: You can provide name-value metadata pairs in the metadata element. These will be stored with the object.
- **Data**: The base 64 encoded form of the data.
- **ContentLength**: The length of the data in bytes.
- **AccessControlList**: An Access Control List for the resource. This element is optional. If omitted, the requester is given FULL_CONTROL access to the object. If the object already exists, the preexisting access control policy is replaced.

Responses

- **ETag**: The entity tag is an MD5 hash of the object that you can use to do conditional fetches of the object using GetObjectExtended. The ETag only reflects changes to the contents of an object, not its metadata.
- **LastModified**: The Amazon S3 timestamp for the saved object.

Access Control

You must have WRITE access to the bucket in order to put objects into the bucket.

Related Resources

- PutObject (SOAP API) (p. 879)
- CopyObject (SOAP API) (p. 880)
PutObject (SOAP API)

Note
SOAP support over HTTP is deprecated, but it is still available over HTTPS. New Amazon S3 features will not be supported for SOAP. We recommend that you use either the REST API or the AWS SDKs.

The PutObject operation adds an object to a bucket. The data for the object is attached as a DIME attachment.

To ensure an object is not corrupted over the network, you can calculate the MD5 of an object, PUT it to Amazon S3, and compare the returned Etag to the calculated MD5 value.

If an object already exists in a bucket, the new object will overwrite it because Amazon S3 stores the last write request. However, Amazon S3 is a distributed system. If Amazon S3 receives multiple write requests for the same object nearly simultaneously, all of the objects might be stored, even though only one wins in the end. Amazon S3 does not provide object locking; if you need this, make sure to build it into your application layer.

Example
This example puts some data and metadata in the "Nelson" object of the "quotes" bucket, give a user (usually the owner) FULL_CONTROL access to the object, and make the object readable by anonymous parties. In this sample, the actual attachment is not shown.

Sample Request

```xml
<PutObject xmlns="http://doc.s3.amazonaws.com/2006-03-01">
  <Bucket>quotes</Bucket>
  <Key>Nelson</Key>
  <Metadata>
    <Name>Content-Type</Name>
    <Value>text/plain</Value>
  </Metadata>
  <Metadata>
    <Name>family</Name>
    <Value>Muntz</Value>
  </Metadata>
  <ContentLength>5</ContentLength>
  <AccessControlList>
    <Grant>
      <Grantee xsi:type="CanonicalUser">
        <ID>a9a7b886d6241bf9b1c61be666e9</ID>
        <DisplayName>chriscustomer</DisplayName>
      </Grantee>
      <Permission>FULL_CONTROL</Permission>
    </Grant>
    <Grant>
      <Grantee xsi:type="Group">
        <URI>http://acs.amazonaws.com/groups/global/AllUsers</URI>
      </Grantee>
      <Permission>READ</Permission>
    </Grant>
  </AccessControlList>
  <AWSAccessKeyId>AKIAIOSFODNN7EXAMPLE</AWSAccessKeyId>
  <Timestamp>2007-05-11T12:00:00.183Z</Timestamp>
  <Signature>Iuyz3d3P0aTou39dzbqaEXAMPLE=</Signature>
</PutObject>
```

Sample Response

```xml
  API Version 2006-03-01
</PutObjectResponse>
```
Elements

- **Bucket**: The bucket in which to add the object.
- **Key**: The key to assign to the object.
- **Metadata**: You can provide name-value metadata pairs in the metadata element. These will be stored with the object.
- **ContentLength**: The length of the data in bytes.
- **AccessControlList**: An Access Control List for the resource. This element is optional. If omitted, the requester is given **FULL_CONTROL** access to the object. If the object already exists, the preexisting Access Control Policy is replaced.

Responses

- **ETag**: The entity tag is an MD5 hash of the object that you can use to do conditional fetches of the object using **GetObjectExtended**. The ETag only reflects changes to the contents of an object, not its metadata.
- **LastModified**: The Amazon S3 timestamp for the saved object.

Access Control

To put objects into a bucket, you must have **WRITE** access to the bucket.

Related Resources

- **CopyObject (SOAP API) (p. 880)**

CopyObject (SOAP API)

**Note**

SOAP support over HTTP is deprecated, but it is still available over HTTPS. New Amazon S3 features will not be supported for SOAP. We recommend that you use either the REST API or the AWS SDKs.

Description

The CopyObject operation creates a copy of an object when you specify the key and bucket of a source object and the key and bucket of a target destination.

When copying an object, you can preserve all metadata (default) or specify new metadata. However, the ACL is not preserved and is set to **private** for the user making the request. To override the default ACL setting, specify a new ACL when generating a copy request. For more information, see Using ACLs.

All copy requests must be authenticated. Additionally, you must have **read** access to the source object and **write** access to the destination bucket. For more information, see Using Auth Access.

To only copy an object under certain conditions, such as whether the Etag matches or whether the object was modified before or after a specified date, use the request parameters...
CopySourceIfUnmodifiedSince, CopyIfUnmodifiedSince, CopySourceIfMatch, or CopySourceIfNoneMatch.

**Note**
You might need to configure the SOAP stack socket timeout for copying large objects.

**Request Syntax**

```xml
<CopyObject xmlns="http://bucket_name.s3.amazonaws.com/2006-03-01">
  <SourceBucket>source_bucket</SourceBucket>
  <SourceObject>source_object</SourceObject>
  <DestinationBucket>destination_bucket</DestinationBucket>
  <DestinationObject>destination_object</DestinationObject>
  <MetadataDirective>{REPLACE | COPY}</MetadataDirective>
  <Metadata>
    <Name>metadata_name</Name>
    <Value>metadata_value</Value>
  </Metadata>
  ...
  <AccessControlList>
    <Grant>
      <Grantee xsi:type="user_type">
        <ID>user_id</ID>
        <DisplayName>display_name</DisplayName>
      </Grantee>
      <Permission>permission</Permission>
    </Grant>
    ...
  </AccessControlList>
  <CopySourceIfMatch>etag</CopySourceIfMatch>
  <CopySourceIfNoneMatch>etag</CopySourceIfNoneMatch>
  <CopySourceIfModifiedSince>date_time</CopySourceIfModifiedSince>
  <CopySourceIfUnmodifiedSince>date_time</CopySourceIfUnmodifiedSince>
  <AWSAccessKeyId>AWSAccessKeyId</AWSAccessKeyId>
  <Timestamp>Timestamp</Timestamp>
  <Signature>Signature</Signature>
</CopyObject>
```

**Request Parameters**

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
<th>Required</th>
</tr>
</thead>
<tbody>
<tr>
<td>SourceBucket</td>
<td>The name of the source bucket.</td>
<td>Yes</td>
</tr>
<tr>
<td></td>
<td>Type: String</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Default: None</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Constraints: A valid source bucket.</td>
<td></td>
</tr>
<tr>
<td>SourceKey</td>
<td>The key name of the source object.</td>
<td>Yes</td>
</tr>
<tr>
<td></td>
<td>Type: String</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Default: None</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Constraints: The key for a valid source object</td>
<td></td>
</tr>
<tr>
<td></td>
<td>to which you have READ access.</td>
<td></td>
</tr>
<tr>
<td>DestinationBucket</td>
<td>The name of the destination bucket.</td>
<td>Yes</td>
</tr>
<tr>
<td>Name</td>
<td>Description</td>
<td>Required</td>
</tr>
<tr>
<td>--------------------</td>
<td>-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
<td>----------</td>
</tr>
<tr>
<td>DestinationKey</td>
<td>The key of the destination object.</td>
<td>Yes</td>
</tr>
<tr>
<td>MetadataDirective</td>
<td>Specifies whether the metadata is copied from the source object or replaced with metadata provided in the request.</td>
<td>No</td>
</tr>
<tr>
<td>Metadata</td>
<td>Specifies metadata name-value pairs to set for the object. If MetadataDirective is set to COPY, all metadata is ignored.</td>
<td>No</td>
</tr>
<tr>
<td>AccessControlList</td>
<td>Grants access to users by e-mail addresses or canonical user ID.</td>
<td>No</td>
</tr>
</tbody>
</table>

Type: String  
Default: None  
Constraints: You must have WRITE access to the destination bucket.

Type: String  
Default: None  
Constraints: You must have WRITE access to the destination bucket.

Type: String  
Default: COPY  
Valid values: COPY | REPLACE

Constraints: Values other than COPY or REPLACE will result in an immediate error. You cannot copy an object to itself unless the MetadataDirective header is specified and its value set to REPLACE.

Type: String  
Default: None  
Constraints: None.
<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
<th>Required</th>
</tr>
</thead>
<tbody>
<tr>
<td>CopySourceIfMatch</td>
<td>Copies the object if its entity tag (ETag) matches the specified tag; otherwise return a PreconditionFailed.</td>
<td>No</td>
</tr>
<tr>
<td></td>
<td>Type: String</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Default: None</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Constraints: None. If the Etag does not match, the object is not copied.</td>
<td></td>
</tr>
<tr>
<td>CopySourceIfNoneMatch</td>
<td>Copies the object if its entity tag (ETag) is different than the specified Etag; otherwise returns an error.</td>
<td>No</td>
</tr>
<tr>
<td></td>
<td>Type: String</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Default: None</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Constraints: None.</td>
<td></td>
</tr>
<tr>
<td>CopySourceIfUnmodifiedSince</td>
<td>Copies the object if it hasn't been modified since the specified time; otherwise returns a PreconditionFailed.</td>
<td>No</td>
</tr>
<tr>
<td></td>
<td>Type: dateTime</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Default: None</td>
<td></td>
</tr>
<tr>
<td>CopySourceIfModifiedSince</td>
<td>Copies the object if it has been modified since the specified time; otherwise returns an error.</td>
<td>No</td>
</tr>
<tr>
<td></td>
<td>Type: dateTime</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Default: None</td>
<td></td>
</tr>
</tbody>
</table>

**Response Syntax**

```xml
<CopyObjectResponse xmlns="http://bucket_name.s3.amazonaws.com/2006-03-01">
  <CopyObjectResponse>
    <ETag>"etag"</ETag>
    <LastModified>timestamp</LastModified>
  </CopyObjectResponse>
</CopyObjectResponse>
```

**Response Elements**

Following is a list of response elements.

**Note**

The SOAP API does not return extra whitespace. Extra whitespace is only returned by the REST API.
<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Etag</td>
<td>Returns the etag of the new object. The ETag only reflects changes to the contents of an object, not its metadata. Type: String Ancestor: CopyObjectResult</td>
</tr>
<tr>
<td>LastModified</td>
<td>Returns the date the object was last modified. Type: String Ancestor: CopyObjectResult</td>
</tr>
</tbody>
</table>

For information about general response elements, see [Using REST Error Response Headers](#).

**Special Errors**

There are no special errors for this operation. For information about general Amazon S3 errors, see [List of Error Codes](#) (p. 817).

**Examples**

This example copies the `flotsam` object from the `pacific` bucket to the `jetsam` object of the `atlantic` bucket, preserving its metadata.

**Sample Request**

```xml
<CopyObject xmlns="http://doc.s3.amazonaws.com/2006-03-01">
  <SourceBucket>pacific</SourceBucket>
  <SourceObject>flotsam</SourceObject>
  <DestinationBucket>atlantic</DestinationBucket>
  <DestinationObject>jetsam</DestinationObject>
  <AWSAccessKeyId>AKIAIOSFODNN7EXAMPLE</AWSAccessKeyId>
  <Timestamp>2008-02-18T13:54:10.183Z</Timestamp>
  <Signature>2uyz3d3P0atou39dzbq7RrtSFmw==</Signature>
</CopyObject>
```

**Sample Response**

```xml
<CopyObjectResponse xmlns="http://doc.s3.amazonaws.com/2006-03-01">
  <ETag>"828ef3fda96f00ad9f27c383fc9ac7f"</ETag>
  <LastModified>2008-02-18T13:54:10.183Z</LastModified>
</CopyObjectResponse>
```

This example copies the "tweedledee" object from the `wonderland` bucket to the "tweedledum" object of the `wonderland` bucket, replacing its metadata.

**Sample Request**

```xml
<CopyObject xmlns="http://doc.s3.amazonaws.com/2006-03-01">
  <SourceBucket>wonderland</SourceBucket>
  <SourceObject>tweedledee</SourceObject>
  <DestinationBucket>wonderland</DestinationBucket>
</CopyObject>
```
<DestinationObject>tweedledum</DestinationObject>
<MetadataDirective>REPLACE</MetadataDirective>
<Metadata>
    <Name>Content-Type</Name>
    <Value>text/plain</Value>
</Metadata>
<Metadata>
    <Name>relationship</Name>
    <Value>twins</Value>
</Metadata>
<AWSAccessKeyId>AKIAIOSFODNN7EXAMPLE</AWSAccessKeyId>
<Timestamp>2008-02-18T13:54:10.183Z</Timestamp>
<Signature>Iuyz3d3P0aTou39dzbq7RrtSFmw=</Signature>
</CopyObject>

Sample Response

<CopyObjectResponse xmlns="http://doc.s3.amazonaws.com/2006-03-01">
    <CopyObjectResponse>
        <ETag>"828ef3fd6a96f00ad9f27c383fc9ac7f"</ETag>
        <LastModified>2008-02-18T13:54:10.183Z</LastModified>
    </CopyObjectResponse>
</CopyObjectResponse>

Related Resources

- PutObject (SOAP API) (p. 879)
- PutObjectinline (SOAP API) (p. 877)

GetObject (SOAP API)

Note

SOAP support over HTTP is deprecated, but it is still available over HTTPS. New Amazon S3 features will not be supported for SOAP. We recommend that you use either the REST API or the AWS SDKs.

The GetObject operation returns the current version of an object. If you try to GetObject an object that has a delete marker as its current version, S3 returns a 404 error. You cannot use the SOAP API to retrieve a specified version of an object. To do that, use the REST API. For more information, see Versioning. For more options, use the GetObjectExtended (SOAP API) (p. 889) operation.

Example

This example gets the "Nelson" object from the "quotes" bucket.

Sample Request

<GetObject xmlns="http://doc.s3.amazonaws.com/2006-03-01">
    <Bucket>quotes</Bucket>
    <Key>Nelson</Key>
    <GetMetadata>true</GetMetadata>
    <GetData>true</GetData>
    <InlineData>true</InlineData>
    <AWSAccessKeyId>AKIAIOSFODNN7EXAMPLE</AWSAccessKeyId>
    <Timestamp>2006-03-01T12:00:00.183Z</Timestamp>
    <Signature>Iuyz3d3P0aTou39dzbq7RrtSFmw=</Signature>
</GetObject>

Sample Response
Amazon Simple Storage Service API Reference
Operations on Objects (SOAP API)

Elements

- **Bucket**: The bucket from which to retrieve the object.
- **Key**: The key that identifies the object.
- **GetMetadata**: The metadata is returned with the object if this is true.
- **GetData**: The object data is returned if this is true.
- **InlineData**: If this is true, then the data is returned, base 64-encoded, as part of the SOAP body of the response. If false, then the data is returned as a SOAP attachment. The InlineData option is not suitable for use with large objects. The system limits this operation to working with 1MB of data or less. A GetObject request with the InlineData flag set will fail with the **InlineDataTooLargeError** status code if the resulting Data parameter would have encoded more than 1MB. To download large objects, consider calling GetObject without setting the InlineData flag, or use the REST API instead.

Returned Elements

- **Metadata**: The name-value paired metadata stored with the object.
- **Data**: If InlineData was true in the request, this contains the base 64 encoded object data.
- **LastModified**: The time that the object was stored in Amazon S3.
- **ETag**: The object's entity tag. This is a hash of the object that can be used to do conditional gets. The ETag only reflects changes to the contents of an object, not its metadata.

Access Control

You can read an object only if you have been granted **READ** access to the object.

SOAP Chunked and Resumable Downloads

To provide GET flexibility, Amazon S3 supports chunked and resumable downloads.

Select from the following:

- For large object downloads, you might want to break them into smaller chunks. For more information, see **Range GETs** (p. 887)
- For GET operations that fail, you can design your application to download the remainder instead of the entire file. For more information, see **REST GET Error Recovery** (p. 889)
Range GETs

For some clients, you might want to break large downloads into smaller downloads. To break a GET into smaller units, use Range.

Before you can break a GET into smaller units, you must determine its size. For example, the following request gets the size of the bigfile object.

```
<ListBucket xmlns="http://doc.s3.amazonaws.com/2006-03-01">
  <Bucket>bigbucket</Bucket>
  <Prefix>bigfile</Prefix>
  <MaxKeys>1</MaxKeys>
  <AWSAccessKeyId>AKIAIOSFODNN7EXAMPLE</AWSAccessKeyId>
  <Timestamp>2006-03-01T12:00:00.183Z</Timestamp>
  <Signature>Iuyz3d3P0aTou39dzbqaEXAMPLE=</Signature>
</ListBucket>
```

Amazon S3 returns the following response.

```
<ListBucketResult xmlns="http://s3.amazonaws.com/doc/2006-03-01">
  <Name>quotes</Name>
  <Prefix>N</Prefix>
  <MaxKeys>1</MaxKeys>
  <IsTruncated>false</IsTruncated>
  <Contents>
    <Key>bigfile</Key>
    <LastModified>2006-01-01T12:00:00.000Z</LastModified>
    <ETag>&quot;828ef3fdfa96f00ad9f27c383fc9ac7f&quot;</ETag>
    <Size>2023276</Size>
    <StorageClass>STANDARD</StorageClass>
    <Owner>
      <ID>bcaf1ffd86f41161ca5fb16fd081034f</ID>
      <DisplayName>bigfile</DisplayName>
    </Owner>
  </Contents>
</ListBucketResult>
```

Following is a request that downloads the first megabyte from the bigfile object.

```
<GetObject xmlns="http://doc.s3.amazonaws.com/2006-03-01">
  <Bucket>bigbucket</Bucket>
  <Key>bigfile</Key>
  <GetMetadata>true</GetMetadata>
  <GetData>true</GetData>
  <InlineData>true</InlineData>
  <ByteRangeStart>0</ByteRangeStart>
  <ByteRangeEnd>1048576</ByteRangeEnd>
  <AWSAccessKeyId>AKIAIOSFODNN7EXAMPLE</AWSAccessKeyId>
  <Timestamp>2006-03-01T12:00:00.183Z</Timestamp>
  <Signature>Iuyz3d3P0aTou39dzbqaEXAMPLE=</Signature>
</GetObject>
```

Amazon S3 returns the first megabyte of the file and the Etag of the file.

```
  <Status>
    <Code>200</Code>
    <Description>OK</Description>
  </Status>
  <Metadata>
```

API Version 2006-03-01

887
To ensure the file did not change since the previous portion was downloaded, specify the IfMatch element. Although the IfMatch element is not required, it is recommended for content that is likely to change.

The following is a request that gets the remainder of the file, using the IfMatch request header.

```
<GetObject xmlns="http://doc.s3.amazonaws.com/2006-03-01">
  <Bucket>bigbucket</Bucket>
  <Key>bigfile</Key>
  <GetMetadata>true</GetMetadata>
  <GetData>true</GetData>
  <InlineData>true</InlineData>
  <ByteRangeStart>10485761</ByteRangeStart>
  <ByteRangeEnd>2023276</ByteRangeEnd>
  <IfMatch>"828ef3fdafa96f00a8f27c383fc9ac7f"</IfMatch>
  <AWSAccessKeyId>AKIAIOSFODNN7EXAMPLE</AWSAccessKeyId>
  <Timestamp>2006-03-01T12:00:00.183Z</Timestamp>
  <Signature>Iuyz3d3P0aTou39dzbqaEXAMPLE=</Signature>
</GetObject>
```

Amazon S3 returns the following response and the remainder of the file.

```
  <GetObjectResponse>
    <Status>
      <Code>200</Code>
      <Description>OK</Description>
    </Status>
    <Metadata>
      <Name>Content-Type</Name>
      <Value>text/plain</Value>
    </Metadata>
    <Metadata>
      <Name>family</Name>
      <Value>Muntz</Value>
    </Metadata>
    <Data>--remainder of bigfile--</Data>
    <LastModified>2006-01-01T12:00:00.000Z</LastModified>
    <ETag>"828ef3fdafa96f00a8f27c383fc9ac7f"</ETag>
  </GetObjectResponse>
</GetObjectResponse>
```

**Versioned GetObject**

The following request returns the specified version of the object in the bucket.

```
<GetObject xmlns="http://doc.s3.amazonaws.com/2006-03-01">
  <Bucket>quotes</Bucket>
  <Key>Nelson</Key>
  <VersionId>1</VersionId>
  <IfMatch>"828ef3fdafa96f00a8f27c383fc9ac7f"</IfMatch>
  <AWSAccessKeyId>AKIAIOSFODNN7EXAMPLE</AWSAccessKeyId>
  <Timestamp>2006-03-01T12:00:00.183Z</Timestamp>
  <Signature>Iuyz3d3P0aTou39dzbqaEXAMPLE=</Signature>
</GetObject>
```

Amazon S3 returns the following response and the remainder of the file.

```
  <GetObjectResponse>
    <Status>
      <Code>200</Code>
      <Description>OK</Description>
    </Status>
    <Metadata>
      <Name>Content-Type</Name>
      <Value>text/plain</Value>
    </Metadata>
    <Metadata>
      <Name>family</Name>
      <Value>"Muntz"</Value>
    </Metadata>
    <Data>--remainder of bigfile--</Data>
    <LastModified>2006-01-01T12:00:00.000Z</LastModified>
    <ETag>"828ef3fdafa96f00a8f27c383fc9ac7f"</ETag>
  </GetObjectResponse>
</GetObjectResponse>
```
<GetMetadata>true</GetMetadata>
<Data>true</Data>
<InlineData>true</InlineData>
<AWSAccessKeyId>AKIAIOSFODNN7EXAMPLE</AWSAccessKeyId>
<Timestamp>2006-03-01T12:00:00.183Z</Timestamp>
<Signature>Iuyz3d3P0aTou39dzbqaEXAMPLE=</Signature>
</GetObject>

Sample Response

<GetObjectResponse>
<Status>
<Code>200</Code>
<Description>OK</Description>
</Status>
<Metadata>
<Name>Content-Type</Name>
<Value>text/plain</Value>
</Metadata>
<Metadata>
<Name>family</Name>
<Value>Muntz</Value>
</Metadata>
<Data>aGEtaGE=</Data>
<LastModified>2006-01-01T12:00:00.000Z</LastModified>
<ETag>"828ef3fdfa96f00ad9f27c383fc9ac7f"</ETag>
</GetObjectResponse>
</GetObjectResponse>

REST GET Error Recovery

If an object GET fails, you can get the rest of the file by specifying the range to download. To do so, you must get the size of the object using ListBucket and perform a range GET on the remainder of the file. For more information, see GetObjectExtended (SOAP API) (p. 889).

Related Resources

Operations on Objects (SOAP API) (p. 876)
GetObjectExtended (SOAP API)

Note
SOAP support over HTTP is deprecated, but it is still available over HTTPS. New Amazon S3 features will not be supported for SOAP. We recommend that you use either the REST API or the AWS SDKs.

GetObjectExtended is exactly like GetObject (SOAP API) (p. 885), except that it supports the following additional elements that can be used to accomplish much of the same functionality provided by HTTP GET headers (go to http://www.w3.org/Protocols/rfc2616/rfc2616-sec14.html).

GetObjectExtended supports the following elements in addition to those supported by GetObject:

- **ByteRangeStart, ByteRangeEnd**: These elements specify that only a portion of the object data should be retrieved. They follow the behavior of the HTTP byte ranges (go to http://www.w3.org/Protocols/rfc2616/rfc2616-sec14.html#sec14.35).
- **IfModifiedSince**: Return the object only if the object's timestamp is later than the specified timestamp. (http://www.w3.org/Protocols/rfc2616/rfc2616-sec14.html#sec14.25)
- **IfUnmodifiedSince**: Return the object only if the object's timestamp is earlier than or equal to the specified timestamp. (go to http://www.w3.org/Protocols/rfc2616/rfc2616-sec14.html#sec14.28)
• IfMatch: Return the object only if its ETag matches the supplied tag(s). (go to http://www.w3.org/Protocols/rfc2616/rfc2616-sec14.html#sec14.24)

• IfNoneMatch: Return the object only if its ETag does not match the supplied tag(s). (go to http://www.w3.org/Protocols/rfc2616/rfc2616-sec14.html#sec14.26)

• ReturnCompleteObjectOnConditionFailure: ReturnCompleteObjectOnConditionFailure: If true, then if the request includes a range element and one or both of IfUnmodifiedSince/IfMatch elements, and the condition fails, return the entire object rather than a fault. This enables the If-Range functionality (go to http://www.w3.org/Protocols/rfc2616/rfc2616-sec14.html#sec14.27).

DeleteObject (SOAP API)

Note
SOAP support over HTTP is deprecated, but it is still available over HTTPS. New Amazon S3 features will not be supported for SOAP. We recommend that you use either the REST API or the AWS SDKs.

The DeleteObject operation removes the specified object from Amazon S3. Once deleted, there is no method to restore or undelete an object.

Note
If you delete an object that does not exist, Amazon S3 will return a success (not an error message).

Example
This example deletes the "Nelson" object from the "quotes" bucket.

Sample Request

```xml
<DeleteObject xmlns="http://doc.s3.amazonaws.com/2006-03-01">
  <Bucket>quotes</Bucket>
  <Key>Nelson</Key>
  <AWSAccessKeyId>AKIAIOSFODNN7EXAMPLE</AWSAccessKeyId>
  <Timestamp>2006-03-01T12:00:00.183Z</Timestamp>
  <Signature>Iuyz3d3P0aTou39dzbqaEXAMPLE=</Signature>
</DeleteObject>
```

Sample Response

```xml
<DeleteObjectResponse xmlns="http://s3.amazonaws.com/doc/2006-03-01">
  <DeleteObjectResponse>
    <Code>200</Code>
    <Description>OK</Description>
  </DeleteObjectResponse>
</DeleteObjectResponse>
```

Elements

• Bucket: The bucket that holds the object.
• Key: The key that identifies the object.

Access Control

You can delete an object only if you have WRITE access to the bucket, regardless of who owns the object or what rights are granted to it.
GetObjectAccessControlPolicy (SOAP API)

Note
SOAP support over HTTP is deprecated, but it is still available over HTTPS. New Amazon S3 features will not be supported for SOAP. We recommend that you use either the REST API or the AWS SDKs.

The GetObjectAccessControlPolicy operation fetches the access control policy for an object.

Example

This example retrieves the access control policy for the "Nelson" object from the "quotes" bucket.

Sample Request

```xml
  <Bucket>quotes</Bucket>
  <Key>Nelson</Key>
  <AWSAccessKeyId>AKIAIOSFODNN7EXAMPLE</AWSAccessKeyId>
  <Timestamp>2006-03-01T12:00:00.183Z</Timestamp>
  <Signature>Iuyz3d3P0aTou39dzbqaEXAMPLE=</Signature>
</GetObjectAccessControlPolicy>
```

Sample Response

```xml
<AccessControlPolicy>
  <Owner>
    <ID>a9a7b886d6fd24a541bf9b1c61be666e9</ID>
    <DisplayName>chriscustomer</DisplayName>
  </Owner>
  <AccessControlList>
    <Grant>
      <Grantee xsi:type="CanonicalUser">
        <ID>a9a7b841bf9b1c61be666e9</ID>
        <DisplayName>chriscustomer</DisplayName>
      </Grantee>
      <Permission>FULL_CONTROL</Permission>
    </Grant>
    <Grant>
      <Grantee xsi:type="Group">
        <URI>http://acs.amazonaws.com/groups/global/AllUsers</URI>
      </Grantee>
      <Permission>READ</Permission>
    </Grant>
  </AccessControlList>
</AccessControlPolicy>
```

Response Body

The response contains the access control policy for the bucket. For an explanation of this response, SOAP Access Policy.

Access Control

You must have READ_ACP rights to the object in order to retrieve the access control policy for an object.

SetObjectAccessControlPolicy (SOAP API)

Note
SOAP support over HTTP is deprecated, but it is still available over HTTPS. New Amazon S3 features will not be supported for SOAP. We recommend that you use either the REST API or the AWS SDKs.
The `SetObjectAccessControlPolicy` operation sets the access control policy for an existing object. If successful, the previous access control policy for the object is entirely replaced with the specified access control policy.

**Example**

This example gives the specified user (usually the owner) FULL_CONTROL access to the "Nelson" object from the "quotes" bucket.

**Sample Request**

```xml
  <Bucket>quotes</Bucket>
  <Key>Nelson</Key>
  <AccessControlList>
    <Grant>
      <Grantee xsi:type="CanonicalUser">
        <ID>a9a7b86d6fd24a52fe8ca5bef65f89a64e0193f23000e241bf9b1c61be666e9</ID>
        <DisplayName>chriscustomer</DisplayName>
      </Grantee>
      <Permission>FULL_CONTROL</Permission>
    </Grant>
  </AccessControlList>
  <AWSAccessKeyId>AKIAIOSFODNN7EXAMPLE</AWSAccessKeyId>
  <Timestamp>2006-03-01T12:00:00.183Z</Timestamp>
  <Signature>Iuyz3d3P0aTou39dzbqaEXAMPLE=</Signature>
</SetObjectAccessControlPolicy>
```

**Sample Response**

```xml
  <SetObjectAccessControlPolicyResponse>
    <Code>200</Code>
    <Description>OK</Description>
  </SetObjectAccessControlPolicyResponse>
</SetObjectAccessControlPolicyResponse>
```

**Access Control**

You must have WRITE_ACP rights to the object in order to set the access control policy for a bucket.

**SOAP Error Responses**

**Note**

SOAP support over HTTP is deprecated, but it is still available over HTTPS. New Amazon S3 features will not be supported for SOAP. We recommend that you use either the REST API or the AWS SDKs.

In SOAP, an error result is returned to the client as a SOAP fault, with the HTTP response code 500. If you do not receive a SOAP fault, then your request was successful. The Amazon S3 SOAP fault code is comprised of a standard SOAP 1.1 fault code (either "Server" or "Client") concatenated with the Amazon S3-specific error code. For example: "Server.InternalError" or "Client.NoSuchBucket". The SOAP fault string element contains a generic, human readable error message in English. Finally, the SOAP fault detail element contains miscellaneous information relevant to the error.

For example, if you attempt to delete the object "Fred", which does not exist, the body of the SOAP response contains a "NoSuchKey" SOAP fault.

The following example shows a sample SOAP error response.
The following table explains the SOAP error response elements:

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Detail</td>
<td>Container for the key involved in the error</td>
</tr>
<tr>
<td></td>
<td>Type: Container</td>
</tr>
<tr>
<td></td>
<td>Ancestor: Body.Fault</td>
</tr>
<tr>
<td>Fault</td>
<td>Container for error information.</td>
</tr>
<tr>
<td></td>
<td>Type: Container</td>
</tr>
<tr>
<td></td>
<td>Ancestor: Body</td>
</tr>
<tr>
<td>Faultcode</td>
<td>The fault code is a string that uniquely identifies an error condition.</td>
</tr>
<tr>
<td></td>
<td>It is meant to be read and understood by programs that detect and handle</td>
</tr>
<tr>
<td></td>
<td>errors by type. For more information, see List of Error Codes (p. 817).</td>
</tr>
<tr>
<td></td>
<td>Type: String</td>
</tr>
<tr>
<td></td>
<td>Ancestor: Body.Fault</td>
</tr>
<tr>
<td>Faultstring</td>
<td>The fault string contains a generic description of the error condition in</td>
</tr>
<tr>
<td></td>
<td>English. It is intended for a human audience. Simple programs display the</td>
</tr>
<tr>
<td></td>
<td>message directly to the end user if they encounter an error condition they</td>
</tr>
<tr>
<td></td>
<td>don't know how or don't care to handle. Sophisticated programs with more</td>
</tr>
<tr>
<td></td>
<td>exhaustive error handling and proper internationalization are more likely to</td>
</tr>
<tr>
<td></td>
<td>ignore the fault string.</td>
</tr>
<tr>
<td></td>
<td>Type: String</td>
</tr>
<tr>
<td></td>
<td>Ancestor: Body.Fault</td>
</tr>
<tr>
<td>Key</td>
<td>Identifies the key involved in the error</td>
</tr>
<tr>
<td></td>
<td>Type: String</td>
</tr>
<tr>
<td></td>
<td>Ancestor: Body.Fault</td>
</tr>
</tbody>
</table>

**Appendix: Lifecycle Configuration APIs (Deprecated)**

Bucket lifecycle configuration is updated to support filters based on object tags. That is, you can now specify a rule that specifies key name prefix, one or more object tags, or both to select a subset of...
objects to which the rule applies. The APIs have been updated accordingly. The following topics describes the prior version of the PUT and GET bucket lifecycle operations for backward compatibility.

Topics
- PUT Bucket lifecycle (Deprecated) (p. 895)
- GET Bucket lifecycle (Deprecated) (p. 905)
PUT Bucket lifecycle (Deprecated)

Description

Important
For an updated version of this API, see PutBucketLifecycleConfiguration (p. 292). This version has been deprecated. Existing lifecycle configurations will work. For new lifecycle configurations, use the updated API.

Creates a new lifecycle configuration for the bucket or replaces an existing lifecycle configuration. For information about lifecycle configuration, see Object Lifecycle Management in the Amazon Simple Storage Service Developer Guide.

Permissions

By default, all Amazon S3 resources, including buckets, objects, and related subresources (for example, lifecycle configuration and website configuration) are private. Only the resource owner, the AWS account that created the resource, can access it. The resource owner can optionally grant access permissions to others by writing an access policy. For this operation, users must get the s3:PutLifecycleConfiguration permission.

You can also explicitly deny permissions. Explicit denial also supersedes any other permissions. If you want to prevent users or accounts from removing or deleting objects from your bucket, you must deny them permissions for the following actions:

- s3:DeleteObject
- s3:DeleteObjectVersion
- s3:PutLifecycleConfiguration

For more information about permissions, see Managing Access Permissions to Your Amazon S3 Resources in the Amazon Simple Storage Service Developer Guide.

Requests

Syntax

```
PUT /?lifecycle HTTP/1.1
Host: bucketname.s3.amazonaws.com
Content-Length: length
Date: date
Authorization: authorization string
Content-MD5: MD5

Lifecycle configuration in the request body
```

For details about authorization strings, see Authenticating Requests (AWS Signature Version 4) (p. 733).

Request Parameters

This implementation of the operation does not use request parameters.

Request Headers

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
<th>Required</th>
</tr>
</thead>
<tbody>
<tr>
<td>Content-MD5</td>
<td>The base64-encoded 128-bit MD5 digest of the data. You must use this header as a message</td>
<td>Yes</td>
</tr>
</tbody>
</table>

API Version 2006-03-01

895
The Name column shows the parameter name, the Description column shows the parameter description, and the Required column shows whether the parameter is required.

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
<th>Required</th>
</tr>
</thead>
<tbody>
<tr>
<td>integrity check</td>
<td>integrity check to verify that the request body was not corrupted in transit. For more information, see RFC 1864.</td>
<td></td>
</tr>
<tr>
<td>Rule ID</td>
<td>Rule ID</td>
<td></td>
</tr>
<tr>
<td>Prefix</td>
<td>Prefix</td>
<td></td>
</tr>
<tr>
<td>Status</td>
<td>Status</td>
<td></td>
</tr>
<tr>
<td>Transition</td>
<td>Transition</td>
<td></td>
</tr>
<tr>
<td>Date</td>
<td>Date</td>
<td></td>
</tr>
<tr>
<td>Storage Class</td>
<td>Storage Class</td>
<td></td>
</tr>
<tr>
<td>Days</td>
<td>Days</td>
<td></td>
</tr>
<tr>
<td>Value</td>
<td>Value</td>
<td></td>
</tr>
<tr>
<td>Noncurrent Version Transition</td>
<td>Noncurrent Version Transition</td>
<td></td>
</tr>
<tr>
<td>Noncurrent Days</td>
<td>Noncurrent Days</td>
<td></td>
</tr>
<tr>
<td>Storage Class</td>
<td>Storage Class</td>
<td></td>
</tr>
<tr>
<td>Noncurrent Version Expiration</td>
<td>Noncurrent Version Expiration</td>
<td></td>
</tr>
<tr>
<td>Noncurrent Days</td>
<td>Noncurrent Days</td>
<td></td>
</tr>
<tr>
<td>Storage Class</td>
<td>Storage Class</td>
<td></td>
</tr>
<tr>
<td>Noncurrent Days</td>
<td>Noncurrent Days</td>
<td></td>
</tr>
</tbody>
</table>

**Request Body**

In the request, you specify the lifecycle configuration in the request body. The lifecycle configuration is specified as XML. The following is an example of a basic lifecycle configuration. It specifies one rule. The *Prefix* in the rule identifies objects to which the rule applies. The rule also specifies two actions (*Transition* and *Expiration*). Each action specifies a timeline when Amazon S3 should perform the action. The *Status* indicates whether the rule is enabled or disabled.

```
<LifecycleConfiguration>
  <Rule>
    <ID>sample-rule</ID>
    <Prefix>key-prefix</Prefix>
    <Status>rule-status</Status>
    <Transition>
      <Date>value</Date>
      <StorageClass>storage class</StorageClass>
    </Transition>
    <Expiration>
      <Days>value</Days>
    </Expiration>
  </Rule>
</LifecycleConfiguration>
```

If the state of your bucket is versioning-enabled or versioning-suspended, you can have many versions of the same object: one current version and zero or more noncurrent versions. The following lifecycle configuration specifies the actions (*Noncurrent Version Transition*, *Noncurrent Version Expiration*) that are specific to noncurrent object versions.

```
<LifecycleConfiguration>
  <Rule>
    <ID>sample-rule</ID>
    <Prefix>key-prefix</Prefix>
    <Status>rule-status</Status>
    <NoncurrentVersionTransition>
      <NoncurrentDays>value</NoncurrentDays>
      <StorageClass>storage class</StorageClass>
    </NoncurrentVersionTransition>
    <NoncurrentVersionExpiration>
      <NoncurrentDays>value</NoncurrentDays>
    </NoncurrentVersionExpiration>
  </Rule>
</LifecycleConfiguration>
```

You can use the multipart upload API to upload large objects in parts. For more information about multipart uploads, see *Multipart Upload Overview* in the *Amazon Simple Storage Service Developer Guide*. With lifecycle configuration, you can tell Amazon S3 to cancel incomplete multipart uploads, which are identified by the key name prefix specified in the rule, if they don't complete within a specified number of days. When Amazon S3 cancels a multipart upload, it deletes all parts associated with the upload. This ensures that you don't have incomplete multipart uploads that have left parts stored in Amazon S3, so
you don't have to pay storage costs for them. The following is an example lifecycle configuration that specifies a rule with the `AbortIncompleteMultipartUpload` action. This action tells Amazon S3 to cancel incomplete multipart uploads seven days after initiation.

```
<LifecycleConfiguration>
  <Rule>
    <ID>sample-rule</ID>
    <Prefix>SomeKeyPrefix</Prefix>
    <Status>rule-status</Status>
    <AbortIncompleteMultipartUpload>
      <DaysAfterInitiation>7</DaysAfterInitiation>
    </AbortIncompleteMultipartUpload>
  </Rule>
</LifecycleConfiguration>
```

The following table describes the XML elements in the lifecycle configuration.

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
<th>Required</th>
</tr>
</thead>
<tbody>
<tr>
<td>AbortIncompleteMultipartUpload</td>
<td>Container for specifying when an incomplete multipart upload becomes eligible for an abort operation.</td>
<td>Yes, if no other action is specified for the rule</td>
</tr>
<tr>
<td></td>
<td>Child: DaysAfterInitiation</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Type: Container</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Ancestor: Rule</td>
<td></td>
</tr>
<tr>
<td>Date</td>
<td>Date when you want Amazon S3 to take the action. For more information, see Lifecycle Rules: Based on a Specific Date in the Amazon Simple Storage Service Developer Guide.</td>
<td>Yes, if Days and ExpiredObjectDeleteMarker are absent</td>
</tr>
<tr>
<td></td>
<td>The date value must conform to ISO 8601 format. The time is always midnight UTC.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Type: String</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Ancestor: Expiration or Transition</td>
<td></td>
</tr>
<tr>
<td>Days</td>
<td>Specifies the number of days after object creation when the specific rule action takes effect.</td>
<td>Yes, if Date and ExpiredObjectDeleteMarker are absent</td>
</tr>
<tr>
<td></td>
<td>Type: Nonnegative Integer when used with Transition, Positive Integer when used with Expiration</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Ancestor: Expiration, Transition</td>
<td></td>
</tr>
<tr>
<td>DaysAfterInitiation</td>
<td>Specifies the number of days after initiating a multipart upload when the multipart upload must be completed. If it does not complete by the specified number of days, it becomes eligible for an abort operation and Amazon S3 cancels the incomplete multipart upload.</td>
<td>Yes, if a parent tag is specified</td>
</tr>
<tr>
<td></td>
<td>Type: Positive Integer</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Ancestor: AbortIncompleteMultipartUpload</td>
<td></td>
</tr>
<tr>
<td>Name</td>
<td>Description</td>
<td>Required</td>
</tr>
<tr>
<td>-----------------------</td>
<td>---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
<td>-----------------------------------</td>
</tr>
<tr>
<td>Expiration</td>
<td>This action specifies a period in an object's lifetime when Amazon S3 should take the appropriate expiration action. The action Amazon S3 takes depends on whether the bucket is versioning-enabled.</td>
<td>Yes, if no other action is present in the Rule.</td>
</tr>
<tr>
<td></td>
<td>• If versioning has never been enabled on the bucket, Amazon S3 deletes the only copy of the object permanently.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>• If the bucket is versioning-enabled (or versioning is suspended), the action applies only to the current version of the object. A versioning-enabled bucket can have many versions of the same object: one current version and zero or more noncurrent versions.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Instead of deleting the current version, Amazon S3 makes it a noncurrent version by adding a delete marker as the new current version.</td>
<td></td>
</tr>
<tr>
<td></td>
<td><strong>Important</strong></td>
<td></td>
</tr>
<tr>
<td></td>
<td>If a bucket's state is versioning-suspended, Amazon S3 creates a delete marker with version ID null. If you have a version with version ID null, Amazon S3 overwrites that version.</td>
<td></td>
</tr>
<tr>
<td></td>
<td><strong>Note</strong></td>
<td></td>
</tr>
<tr>
<td></td>
<td>To set the expiration for noncurrent objects, use the NoncurrentVersionExpiration action.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Type: Container</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Children: Days or Date</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Ancestor: Rule</td>
<td></td>
</tr>
<tr>
<td>ID</td>
<td>Unique identifier for the rule. The value cannot be longer than 255 characters.</td>
<td>No</td>
</tr>
<tr>
<td></td>
<td>Type: String</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Ancestor: Rule</td>
<td></td>
</tr>
<tr>
<td>LifecycleConfiguration</td>
<td>Container for lifecycle rules. You can add as many as 1000 rules.</td>
<td>Yes</td>
</tr>
<tr>
<td></td>
<td>Type: Container</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Children: Rule</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Ancestor: None</td>
<td></td>
</tr>
<tr>
<td>Name</td>
<td>Description</td>
<td>Required</td>
</tr>
<tr>
<td>------------------------------------</td>
<td>-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
<td>-------------------</td>
</tr>
<tr>
<td>ExpiredObjectDeleteMarker</td>
<td>On a versioned bucket (a versioning-enabled or versioning-suspended bucket), you can add this element in the lifecycle configuration to tell Amazon S3 to delete expired object delete markers. For an example, see Example 8: Removing Expired Object Delete Markers in the Amazon Simple Storage Service Developer Guide. Don't add it to a non-versioned bucket, because that type of bucket cannot include delete markers.</td>
<td>Yes, if Date and Days are absent</td>
</tr>
<tr>
<td></td>
<td>Type: String</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Valid values: true</td>
<td>false (the value false is allowed, but it is no-op, which means that Amazon S3 will not take action)</td>
</tr>
<tr>
<td></td>
<td>Ancestor: Expiration</td>
<td></td>
</tr>
<tr>
<td>NoncurrentDays</td>
<td>Specifies the number of days an object is noncurrent before Amazon S3 can perform the associated action. For information about the noncurrent days calculations, see How Amazon S3 Calculates When an Object Became Noncurrent in the Amazon Simple Storage Service Developer Guide.</td>
<td>Yes</td>
</tr>
<tr>
<td></td>
<td>Type: Nonnegative Integer when used with NoncurrentVersionTransition, Positive Integer when used with NoncurrentVersionExpiration</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Ancestor: NoncurrentVersionExpiration or NoncurrentVersionTransition</td>
<td></td>
</tr>
<tr>
<td>NoncurrentVersionExpiration</td>
<td>Specifies when noncurrent object versions expire. Upon expiration, Amazon S3 permanently deletes the noncurrent object versions. Set this lifecycle configuration action on a bucket that has versioning enabled (or suspended) to tell Amazon S3 to delete noncurrent object versions at a specific period in the object's lifetime.</td>
<td>Yes, if no other action is present in the Rule</td>
</tr>
<tr>
<td></td>
<td>Type: Container</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Children: NoncurrentDays</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Ancestor: Rule</td>
<td></td>
</tr>
<tr>
<td>Name</td>
<td>Description</td>
<td>Required</td>
</tr>
<tr>
<td>-----------------------------</td>
<td>-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
<td>-----------------------------------</td>
</tr>
</tbody>
</table>
| NoncurrentVersionTransition | Container for the transition rule that describes when noncurrent objects transition to the STANDARD_IA, ONEZONE_IA, or GLACIER storage class. If your bucket is versioning-enabled (or if versioning is suspended), you can set this action to tell Amazon S3 to transition noncurrent object versions at a specific period in the object's lifetime. Type: Container  
   Children: NoncurrentDays and StorageClass  
   Ancestor: Rule | Yes, if no other action is present in the Rule |
| Prefix                      | Object key prefix that identifies one or more objects to which the rule applies. Type: String  
   Ancestor: Rule | Yes |
| Rule                        | Container for a lifecycle rule. A lifecycle configuration can contain as many as 1000 rules. Type: Container  
   Ancestor: LifecycleConfiguration | Yes |
| Status                      | If enabled, Amazon S3 executes the rule as scheduled. If it is disabled, Amazon S3 ignores the rule. Type: String  
   Ancestor: Rule  
   Valid values: Enabled, Disabled | Yes |
| StorageClass                | Specifies the Amazon S3 storage class to which you want the object to transition. Type: String  
   Ancestor: Transition and NoncurrentVersionTransition  
   Valid values: STANDARD_IA | ONEZONE_IA | GLACIER | Yes  
   This element is required only if you specify one or both its ancestors. |
<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
<th>Required</th>
</tr>
</thead>
</table>
| Transition  | This action specifies a period in the objects' lifetime when Amazon S3 should transition them to the STANDARD_IA, ONEZONE_IA, or GLACIER storage class. When this action is in effect, what Amazon S3 does depends on whether the bucket is versioning-enabled.  
• If versioning has never been enabled on the bucket, Amazon S3 transitions the only copy of the object to the specified storage class.  
• If your bucket is versioning-enabled (or versioning is suspended), Amazon S3 transitions only the current versions of objects identified in the rule.  
  **Note**  
  A versioning-enabled bucket can have many versions of an object. This action has no effect on noncurrent object versions. To transition noncurrent objects, you must use the NoncurrentVersionTransition action. |
|             | Yes, if no other action is present in the Rule                                                                                                                                                    |

**Type:** Container  
**Children:** Days or Date, and StorageClass  
**Ancestor:** Rule

### Responses

**Response Headers**

This implementation of the operation uses only response headers that are common to most responses. For more information, see Common Response Headers (p. 814).

**Response Elements**

This implementation of the operation does not return response elements.

**Special Errors**

This implementation of the operation does not return special errors. For general information about Amazon S3 errors and a list of error codes, see Error Responses (p. 816).

### Examples

**Example 1: Add Lifecycle Configuration to a Bucket That Is Not Versioning-enabled**

The following lifecycle configuration specifies two rules, each with one action.
• The Transition action tells Amazon S3 to transition objects with the "documents/" prefix to the GLACIER storage class 30 days after creation.
• The Expiration action tells Amazon S3 to delete objects with the "logs/" prefix 365 days after creation.

```
<LifecycleConfiguration>
  <Rule>
    <ID>id1</ID>
    <Prefix>documents/</Prefix>
    <Status>Enabled</Status>
    <Transition>
      <Days>30</Days>
      <StorageClass>GLACIER</StorageClass>
    </Transition>
  </Rule>
  <Rule>
    <ID>id2</ID>
    <Prefix>logs/</Prefix>
    <Status>Enabled</Status>
    <Expiration>
      <Days>365</Days>
    </Expiration>
  </Rule>
</LifecycleConfiguration>
```

The following is a sample PUT /?lifecycle request that adds the preceding lifecycle configuration to the examplebucket bucket.

```
PUT /?lifecycle HTTP/1.1
Host: examplebucket.s3.amazonaws.com
x-amz-date: Wed, 14 May 2014 02:11:21 GMT
Content-MD5: q6yJDl1kbcAGf3GLY69A==
Authorization: authorization string
Content-Length: 415

<LifecycleConfiguration>
  <Rule>
    <ID>id1</ID>
    <Prefix>documents/</Prefix>
    <Status>Enabled</Status>
    <Transition>
      <Days>30</Days>
      <StorageClass>GLACIER</StorageClass>
    </Transition>
  </Rule>
  <Rule>
    <ID>id2</ID>
    <Prefix>logs/</Prefix>
    <Status>Enabled</Status>
    <Expiration>
      <Days>365</Days>
    </Expiration>
  </Rule>
</LifecycleConfiguration>
```

The following is a sample response.

```
HTTP/1.1 200 OK
x-amz-id-2: r+qR7+nhXtJDDIJ0JJYcd+1j5nM/rUFliiZ/fN6DOs3JUE8NWMLNHXmvPfwMpdc
x-amz-request-id: 9E26D08072A8EF9E
Date: Wed, 14 May 2014 02:11:22 GMT
```

API Version 2006-03-01
Example 2: Add Lifecycle Configuration to a Versioning-enabled Bucket

The following lifecycle configuration specifies two rules, each with one action for Amazon S3 to perform. You specify these actions when your bucket is versioning-enabled or versioning is suspended:

- The **NoncurrentVersionExpiration** action tells Amazon S3 to expire noncurrent versions of objects with the "logs/" prefix 100 days after the objects become noncurrent.
- The **NoncurrentVersionTransition** action tells Amazon S3 to transition noncurrent versions of objects with the "documents/" prefix to the GLACIER storage class 30 days after they become noncurrent.

```xml
<LifecycleConfiguration>
  <Rule>
    <ID>DeleteAfterBecomingNonCurrent</ID>
    <Prefix>logs/</Prefix>
    <Status>Enabled</Status>
    <NoncurrentVersionExpiration>
      <NoncurrentDays>100</NoncurrentDays>
    </NoncurrentVersionExpiration>
  </Rule>
  <Rule>
    <ID>TransitionAfterBecomingNonCurrent</ID>
    <Prefix>documents/</Prefix>
    <Status>Enabled</Status>
    <NoncurrentVersionTransition>
      <NoncurrentDays>30</NoncurrentDays>
      <StorageClass>GLACIER</StorageClass>
    </NoncurrentVersionTransition>
  </Rule>
</LifecycleConfiguration>
```

The following is a sample PUT `/?lifecycle` request that adds the preceding lifecycle configuration to the examplebucket bucket.

```plaintext
PUT /?lifecycle HTTP/1.1
Host: examplebucket.s3.amazonaws.com
x-amz-date: Wed, 14 May 2014 02:21:48 GMT
Content-MD5: 96rxH9mDqVNKkaZddgme==
Authorization: authorization string
Content-Length: 598

<LifecycleConfiguration>
  <Rule>
    <ID>DeleteAfterBecomingNonCurrent</ID>
    <Prefix>logs/</Prefix>
    <Status>Enabled</Status>
    <NoncurrentVersionExpiration>
      <NoncurrentDays>1</NoncurrentDays>
    </NoncurrentVersionExpiration>
  </Rule>
  <Rule>
    <ID>TransitionSoonAfterBecomingNonCurrent</ID>
    <Prefix>documents/</Prefix>
    <Status>Enabled</Status>
    <NoncurrentVersionTransition>
      <NoncurrentDays>0</NoncurrentDays>
      <StorageClass>GLACIER</StorageClass>
    </NoncurrentVersionTransition>
  </Rule>
</LifecycleConfiguration>
```
The following is a sample response.

HTTP/1.1 200 OK
x-amz-id-2: aXQ+KbIrMm0O//3bMdDTw/CnjArwje+J49Hf+j44yRb/Vmb1kgIo5A+PT98Cp/6k07hf+LD2mY=
x-amz-request-id: 02D7EC4C10381EB1
Date: Wed, 14 May 2014 02:21:50 GMT
Content-Length: 0
Server: AmazonS3

Additional Examples

For more examples of transitioning objects to storage classes such as STANDARD_IA or ONEZONE_IA, see Examples of Lifecycle Configuration.

Related Resources

- GetBucketLifecycleConfiguration (p. 113)
- POST Object restore (p. 781)
- By default, a resource owner—in this case, a bucket owner, which is the AWS account that created the bucket—can perform any of the operations. A resource owner can also grant others permission to perform the operation. For more information, see the following topics in the Amazon Simple Storage Service Developer Guide:
  - Specifying Permissions in a Policy
  - Managing Access Permissions to Your Amazon S3 Resources
GET Bucket lifecycle (Deprecated)

Description

Important
For an updated version of this API, see GetBucketLifecycleConfiguration (p. 113). If you configured a bucket lifecycle using the <filter> element, you should see an updated version of this topic. This topic is provided for backward compatibility.

Returns the lifecycle configuration information set on the bucket. For information about lifecycle configuration, go to Object Lifecycle Management in the Amazon Simple Storage Service Developer Guide.

To use this operation, you must have permission to perform the s3:GetLifecycleConfiguration action. The bucket owner has this permission by default. The bucket owner can grant this permission to others. For more information about permissions, see Managing Access Permissions to Your Amazon S3 Resources in the Amazon Simple Storage Service Developer Guide.

Requests

Syntax

GET /?lifecycle HTTP/1.1
Host: bucketname.s3.amazonaws.com
Date: date
Authorization: authorization string (see Authenticating Requests (AWS Signature Version 4))

Request Parameters

This implementation of the operation does not use request parameters.

Request Headers

This implementation of the operation uses only request headers that are common to all operations. For more information, see Common Request Headers (p. 812).

Request Elements

This implementation of the operation does not use request elements.

Responses

Response Headers

This implementation of the operation uses only response headers that are common to most responses. For more information, see Common Response Headers (p. 814).

Response Elements

This implementation of GET returns the following response elements.

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
<th>Required</th>
</tr>
</thead>
<tbody>
<tr>
<td>AbortIncompleteMultipartUpload</td>
<td>Container for specifying when an incomplete multipart upload becomes eligible for an abort operation.</td>
<td>Yes, if no other action is specified for the rule</td>
</tr>
<tr>
<td>Name</td>
<td>Description</td>
<td>Required</td>
</tr>
<tr>
<td>------------------------</td>
<td>--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
<td>---------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td></td>
<td>Child: DaysAfterInitiation</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Type: Container</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Ancestor: Rule</td>
<td></td>
</tr>
<tr>
<td>Date</td>
<td>Date when you want Amazon S3 to take the action. For more information, see Lifecycle Rules: Based on a Specific Date in the Amazon Simple Storage Service Developer Guide. The date value must conform to the ISO 8601 format. The time is always midnight UTC.</td>
<td>Yes, if Days and ExpiredObjectDeleteMarker are absent</td>
</tr>
<tr>
<td></td>
<td>Type: String</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Ancestor: Expiration or Transition</td>
<td></td>
</tr>
<tr>
<td>Days</td>
<td>Specifies the number of days after object creation when the specific rule action takes effect. The object's eligibility time is calculated as creation time + the number of days with the resulting time rounded to midnight UTC of the next day.</td>
<td>Yes, if Date and ExpiredObjectDeleteMarker are absent</td>
</tr>
<tr>
<td></td>
<td>Type: Non-negative Integer when used with Transition, Positive Integer when used with Expiration.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Ancestor: Transition or Expiration</td>
<td></td>
</tr>
<tr>
<td>DaysAfterInitiation</td>
<td>Specifies the number of days after initiating a multipart upload when the multipart upload must be completed. If it does not complete by the specified number of days, it becomes eligible for an abort operation and Amazon S3 cancels the incomplete multipart upload.</td>
<td>Yes, if Date is absent</td>
</tr>
<tr>
<td></td>
<td>Type: Positive Integer</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Ancestor: AbortIncompleteMultipartUpload</td>
<td></td>
</tr>
<tr>
<td>Name</td>
<td>Description</td>
<td>Required</td>
</tr>
<tr>
<td>----------</td>
<td>-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
<td>---------------------------</td>
</tr>
<tr>
<td>Expiration</td>
<td>This action specifies a period in the object's lifetime when Amazon S3 should take the appropriate expiration action. The expiration action occurs only on objects that are eligible according to the period specified in the child Date or Days element. The action Amazon S3 takes depends on whether the bucket is versioning enabled.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>• If versioning has never been enabled on the bucket, Amazon S3 deletes the only copy of the object permanently.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>• Otherwise, if your bucket is versioning-enabled (or versioning is suspended), the action applies only to the current version of the object. Buckets that are versioning-enabled or versioning-suspended can have many versions of the same object: one current version, and zero or more noncurrent versions.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Instead of deleting the current version, Amazon S3 makes it a noncurrent version by adding a delete marker as the new current version.</td>
<td></td>
</tr>
<tr>
<td></td>
<td><strong>Important</strong></td>
<td></td>
</tr>
<tr>
<td></td>
<td>If the state of a bucket is versioning-suspended, Amazon S3 creates a delete marker with version ID null. If you have a version with version ID null, then Amazon S3 overwrites that version.</td>
<td></td>
</tr>
<tr>
<td></td>
<td><strong>Note</strong></td>
<td></td>
</tr>
<tr>
<td></td>
<td>To set the expiration for noncurrent objects, you must use the NoncurrentVersionExpiration action.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Type: Container</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Children: Days or Date</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Ancestor: Rule</td>
<td></td>
</tr>
<tr>
<td>ID</td>
<td>Unique identifier for the rule. The value cannot be longer than 255 characters.</td>
<td>No</td>
</tr>
<tr>
<td></td>
<td>Type: String</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Ancestor: Rule</td>
<td></td>
</tr>
<tr>
<td>Name</td>
<td>Description</td>
<td>Required</td>
</tr>
<tr>
<td>-------------------------</td>
<td>-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
<td>-------------</td>
</tr>
<tr>
<td>LifecycleConfiguration</td>
<td>Container for lifecycle rules. You can add as many as 1000 rules. Type: Container Children: Rule Ancestor: None</td>
<td>Yes</td>
</tr>
<tr>
<td>ExpiredObjectDeleteMarker</td>
<td>On a versioned bucket (versioning-enabled or versioning-suspended bucket), this element indicates whether Amazon S3 will delete any expired object delete markers in the bucket. For an example, go to Example 8: Specify Expiration Action to Remove Expired Object Delete Markers in the Amazon Simple Storage Service Developer Guide. Type: String Valid values: true</td>
<td>Yes, if Date and Days are absent</td>
</tr>
<tr>
<td></td>
<td>Expiration valid values: true</td>
<td></td>
</tr>
<tr>
<td></td>
<td>ExpiredObjectDeleteMarker</td>
<td>Ancestor: Expiration</td>
</tr>
<tr>
<td>NoncurrentDays</td>
<td>Specifies the number of days that an object is noncurrent before Amazon S3 can perform the associated action. For information about calculating noncurrent days, see Lifecycle Rules Based on the Number of Days in the Amazon Simple Storage Service Developer Guide. Type: Nonnegative Integer when used with NoncurrentVersionTransition, Positive Integer when used with NoncurrentVersionExpiration Ancestor: NoncurrentVersionExpiration or NoncurrentVersionTransition</td>
<td>Yes, only if the ancestor is present</td>
</tr>
<tr>
<td>NoncurrentVersionExpiration</td>
<td>Specifies when noncurrent object versions expire. Upon expiration, Amazon S3 permanently deletes the noncurrent object versions. Set this lifecycle configuration action on a bucket that has versioning enabled (or suspended) to request that Amazon S3 delete noncurrent object versions at a specific period in the object's lifetime. Type: Container Children: NoncurrentDays Ancestor: Rule</td>
<td>Yes, if no other action is present in the Rule</td>
</tr>
<tr>
<td>Name</td>
<td>Description</td>
<td>Required</td>
</tr>
<tr>
<td>------------------------------</td>
<td>-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
<td>-----------------------------------------------------------------------------------------------</td>
</tr>
</tbody>
</table>
| NoncurrentVersionTransition  | Container for the transition rule that describes when noncurrent objects transition to the STANDARD_IA, ONEZONE_IA, or the GLACIER storage class. If your bucket is versioning-enabled (or versioning is suspended), you can set this action to request Amazon S3 to transition noncurrent object versions to the GLACIER storage class at a specific period in the object's lifetime. Type: Container  

Children: NoncurrentDays and StorageClass  

Ancestor: Rule                                                                                   | Yes, if no other action is present in the Rule                                               |
| Prefix                        | Object key prefix identifying one or more objects to which the rule applies. Type: String  

Ancestor: Rule                                                                                   | Yes                                                                                         |
| Rule                          | Container for a lifecycle rule. Type: Container  

Ancestor: LifecycleConfiguration                                                              | Yes                                                                                         |
| Status                        | If Enabled, Amazon S3 executes the rule as scheduled. If Disabled, Amazon S3 ignores the rule. Type: String  

Ancestor: Rule  

Valid values: Enabled or Disabled                                                            | Yes                                                                                         |
| StorageClass                  | Specifies the Amazon S3 storage class to which you want to transition the object. Type: String  

Ancestor: Transition and NoncurrentVersionTransition  

Valid values: STANDARD_IA | ONEZONE_IA | GLACIER                                                                 | Yes                                                                                         |
### Transition

This action specifies a period in the objects' lifetime when Amazon S3 should transition them to the STANDARD_IA, ONEZONE_IA, or GLACIER storage class. When this action is in effect, what Amazon S3 does depends on whether the bucket is versioning-enabled.

- If versioning has never been enabled on the bucket, Amazon S3 transitions the only copy of the object to the specified storage class.
- When your bucket is versioning-enabled (or versioning is suspended), Amazon S3 transitions only the current versions of the objects identified in the rule.

**Note**

A versioning-enabled or versioning-suspended bucket can contain many versions of an object. This action has no effect on the noncurrent object versions. To transition noncurrent objects, you must use the NoncurrentVersionTransition action.

Type: Container

Children: Days or Date, and StorageClass

Ancestor: Rule

---

### Special Errors

**Error Code** | **Description** | **HTTP Status Code** | **SOAP Fault Code Prefix**
---|---|---|---
NoSuchLifecycleConfiguration | The lifecycle configuration does not exist. | 404 Not Found | Client

For general information about Amazon S3 errors and a list of error codes, see Error Responses (p. 816).

### Examples

**Example 1: Retrieve a Lifecycle Subresource**

This example is a GET request to retrieve the lifecycle subresource from the specified bucket, and an example response with the returned lifecycle configuration.

**Sample Request**

```
GET /?lifecycle HTTP/1.1
Host: examplebucket.s3.amazonaws.com
```
Sample Response

HTTP/1.1 200 OK
x-amz-id-2: ITnGT1y4RyTmXa3rPi4hklTXouTf0hccUjo0iCFjz6FnfIutBj3M7fPGlWQ2SEWp
x-amz-request-id: 51991C342C575321
Date: Thu, 15 Nov 2012 00:17:23 GMT
Server: AmazonS3
Content-Length: 358

<?xml version="1.0" encoding="UTF-8"?>
<LifecycleConfiguration xmlns="http://s3.amazonaws.com/doc/2006-03-01/">
    <Rule>
        <ID>Archive and then delete rule</ID>
        <Prefix>projectdocs/</Prefix>
        <Status>Enabled</Status>
        <Transition>
            <Days>30</Days>
            <StorageClass>STANDARD_IA</StorageClass>
        </Transition>
        <Transition>
            <Days>365</Days>
            <StorageClass>GLACIER</StorageClass>
        </Transition>
        <Expiration>
            <Days>3650</Days>
        </Expiration>
    </Rule>
</LifecycleConfiguration>

Related Resources

- PutBucketLifecycleConfiguration (p. 292)
- DeleteBucketLifecycle (p. 55)