

Pure Storage FlashBlade Object Store REST API



Version 1.0

March 2019

purestorage.com

© 2019 Pure Storage, Inc. All rights reserved. Pure Storage, Pure1, and the Pure Storage logo are trademarks or registered trademarks of Pure Storage, Inc. in the U.S. and other countries. Other company, product, or service names may be trademarks or service marks of their respective owners. The Pure Storage products described in this documentation are distributed under a license agreement restricting the use, copying, distribution, and decompilation/reverse engineering of the products. The Pure Storage products described in this documentation may only be used in accordance with the terms of the license agreement. No part of this documentation may be reproduced in any form by any means without prior written authorization from Pure Storage, Inc. and its licensors, if any. Pure Storage may make improvements and/or changes in the Pure Storage products and/or the programs described in this documentation at any time without notice. THIS DOCUMENTATION IS PROVIDED "AS IS" AND ALL EXPRESS OR IMPLIED CONDITIONS, REPRESENTATIONS AND WARRANTIES, INCLUDING ANY IMPLIED WARRANTY OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, OR NONINFRINGEMENT, ARE DISCLAIMED, EXCEPT TO THE EXTENT THAT SUCH DISCLAIMERS ARE HELD TO BE LEGALLY INVALID. PURE STORAGE SHALL NOT BE LIABLE FOR INCIDENTAL OR CONSEQUENTIAL DAMAGES IN CONNECTION WITH THE FURNISHING, PERFORMANCE, OR USE OF THIS DOCUMENTATION. THE INFORMATION CONTAINED IN THIS DOCUMENTATION IS SUBJECT TO CHANGE WITHOUT NOTICE.

Introduction.....	5
REST API Overview.....	5
What is REST?.....	5
Flash Blade Object Store REST API.....	7
Authentication	7
Supported Protocols.....	7
Pre-signed URLs	7
Response Codes.....	8
Content Type	11
Amazon S3 REST API Compatibility	12
Supported Bucket and Object Operations	12
Bucket Operations	13
Common Request Headers	13
Common Response Headers	14
DELETE Bucket	14
GET Bucket (List Objects) Version 1.....	15
GET Bucket (List Object) Version 2.....	16
GET Bucket Location	18
List Multipart Uploads.....	19
GET Bucket Object Versions	21
HEAD Bucket	22
PUT Bucket.....	23
PUT Bucket Lifecycle.....	23
DELETE Bucket Lifecycle.....	25
GET Bucket Lifecycle.....	25
PUT Bucket Versioning	27
GET Bucket Versioning	28

Object Operations	29
POST Object	29
DELETE Object	31
GET Object	31
HEAD Object	33
DELETE Multiple Objects	34
PUT Object	36
PUT Object Copy	38
Abort Multipart Upload	40
List Parts	42
Initiate Multipart Upload	46
Complete Multipart Upload	48
Upload Part	51
Upload Part Copy	54

Introduction

This guide describes how to manipulate objects in the Pure Storage FlashBlade object store using the FlashBlade Object Store REST API. Credentialed users can manipulate buckets and objects from a remote host using these REST API requests.

The guide is divided into two sections: Overview and Resources. The Overview section describes high-level concepts while the Resources section contains a comprehensive list of resources and endpoints.

REST API Overview

The REST API provides simple interfaces using HTTP requests to interact with buckets and objects on a FlashBlade object store.

What is REST?

REpresentational State Transfer (REST) uses HTTP/HTTPS requests to interact with resources on FlashBlade. REST is an architectural style that describes the following constraints:

- Uniform interface: Individual resources are identified in requests using URIs resource identifiers. For example:

```
bucket  
bucket/object
```

Clients manipulate resources through request methods such as:

METHOD	DESCRIPTION	EXAMPLE
DELETE	Delete a resource	DELETE bucket/resource
GET	List resources	GET bucket/object
HEAD	Lists resources in the HTTP header	HEAD bucket/object
POST	Create a resource	POST bucket/object
PUT	Modify a resource	PUT bucket/object

- Requests can include additional parameters, either in the URL or as JSON post data. For example, the location parameter is included in the following request to list all bucket locations:

```
GET /?location HTTP/1.1
Host: bucket.s3.amazonaws.com
Date: Tue, 05 Dec 2017 11:35:04 +0000
Authorization: signatureValue
```

- Responses return a response code and data (represented in JSON). For example, the following GET bucket location request returns a response code of 200 (OK) and the location of the bucket in the XML payload of the response:

```
HTTP/1.1 200 OK
Content-Type: application/xml
Content-Length: 151

<?xml version="1.0" encoding="UTF-8"
standalone="yes"?><LocationConstraint
xmlns="http://s3.amazonaws.com/doc/2006-03-01/">on-
prem</LocationConstraint>
```

- Stateless: The necessary state to handle a request is contained in the request itself.
- Cacheable: The responses must implicitly or explicitly define themselves as cacheable.
- Client-server: The uniform interface creates separation between client and server. Clients can be more portable, and servers do not need to know anything about user interface or state.
- Layered system: A client cannot tell if it is connected directly to the server; intermediary servers can be used to improve scalability or enforce Security policies.

Flash Blade Object Store REST API

This API provides simple interfaces for data path actions on buckets and objects in object stores on the FlashBlade.

Authentication

AWS Signature v2, Signature v4, and chunked transfer payloads are supported. Refer to [AWS Signature v2](#) and [Signature v4](#) documentation for details.

Restrictions

Anonymous requests to create buckets are not supported. You must have a valid access key ID to authenticate requests in order to create a bucket.

Supported Protocols

The Flash Blade object store REST API supports both HTTP (preferred) and HTTPS protocols.

Pre-signed URLs

Pre-signed URLs allow you to grant limited access to objects identified in the URLs.

By default, all objects and buckets are private. Pre-signed URLs are useful to allow others the ability to upload a specific object to your bucket without requiring them to have AWS security credentials or permissions.

Object owners can optionally share objects with others by creating a pre-signed URL, using their own security credentials, to allow time-limited permission to download the objects.

Pure Storage currently supports pre-signed URLs with the PUT object and GET object operations.

For complete information about using pre-signed URLs refer to the [Amazon Simple Storage Service Developer Guide](#).

Response Codes

Each REST API request returns a response code. Successful requests return the response code 200 OK. Scripts must be prepared to handle responses with codes other than 200 OK.

NAME	CODE	DESCRIPTION
Access Denied	403	Access denied.
AccountProblem	403	There is a problem with your AWS account that prevents the operation from completing successfully.
AmbiguousGrantByEmailAddress	400	The provided email address is associated with more than one account.
BadDigest	400	The Content-MD5 you specified did not match what we received.
BucketAlreadyExists	409	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.
BucketNotEmpty	409	The bucket you tried to delete is not empty.
CredentialsNotSupported	400	This request does not support credentials.
CrossLocationLoggingProhibited	403	Cross-location logging is not allowed. Buckets in one geographic location cannot log information to a bucket in another location.
EntityTooSmall	400	Your proposed upload is smaller than the minimum allowed object size.
EntityTooLarge	400	Your proposed upload exceeds the maximum allowed object size.
EntityToken	400	The provided token has expired.
InternalServerError	500	We encountered an internal error. Please try again.
InvalidAccessKeyId	403	The access key Id you provided does not exist in our records.
InvalidAddressingHeader	N/A	You must specify the Anonymous role.
InvalidArgument	400	Invalid argument.
InvalidBucketName	400	The specified bucket is not valid.
InvalidBucketState	409	The request is not valid with the current state of the bucket.
InvalidDigest	400	The Content-MD5 you specified is invalid.

NAME	CODE	DESCRIPTION
InvalidEncryptionAlgorithmError	400	The encryption request is invalid. The valid value is AES256.
InvalidLocaionConstraint	400	The specified location restraint is invalid.
InvalidObjectState	403	The operation is invalid for the current state of the object.
InvalidPart	400	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.
InvalidPartOrder	400	The list of parts was not in ascending order. Parts list must be specified in order by part number.
InvalidPayer	403	All access to this object has been disabled.
InvalidPolicyDocument	400	The content of the form does not meet the conditions specified in the policy document.
InvalidRange	416	The requested range cannot be satisfied.
InvalidRequest	400	The request is invalid.
InvalidSecurity	403	The provided security credentials are not valid.
InvalidSOAPRequest	400	The SOAP request body is invalid.
InvalidStorageClass	400	The specified storage class is invalid.
InvalidTargetBucketForLogging	400	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.
InvalidToken	400	The provided token is malformed or otherwise invalid.
InvalidURI	400	Could not parse the specified URI.
KeyTooLong	400	Your key is too long.
MalformedACLError	400	The provided XML was not well-formed or did not validate against our published schema.
MalformedPOSTRequest	400	The body of your POST request is not well-formed multipart/form-data.
MalformedXML	400	The XML you provided was not well-formed or did not validate against our published schema.
MaxMessageLengthExceeded	400	Your request was too big.
MaxPostPreDataLengthExceededError	400	Your POST request fields preceding the upload file were too large.

NAME	CODE	DESCRIPTION
MetadataTooLarge	400	Your metadata headers exceed the maximum allowed metadata size.
MethodNotAllowed	405	The specified method is not allowed against this resource.
MissingAttachment	N/A	A SOAP attachment was expected, but none were returned.
MissingContentLength	411	You must provide the Content-Length HTTP header.
MissingRequestBodyError	400	An empty XML document was sent as a request.
MissingSecurityElement	400	The SOAP 1.1 request is missing a security element.
MissingSecurityHeader	400	Your request is missing a required header.
NoLoggingStatusForKey	400	There is no such thing as a logging status subresource for a key.
NoSuchBucket	404	The specified bucket does not exist.
NoSuchKey	404	The specified key does not exist.
NoSuchLifecycleConfiguration	404	The lifecycle configuration does not exist.
NoSuchUpload	404	The specified multipart upload does not exist. The upload ID might be invalid, or the multipart upload might have been aborted or completed.
NoSuchVersion	404	The version ID specified in the request does not match an existing version.
NotImplemented	501	A header you provided implies functionality that is not implemented.
NotSignedUp	403	Your account is not signed up for the Amazon S3 service. You must sign up before you can use Amazon S3. Sign up at https://aws.amazon.com/s3 .
NoSuchBucketPolicy	404	The specified bucket does not have a bucket policy.
OperationAborted	409	A conflicting conditional operation is currently in progress against this resource. Try again.
PermanentRedirect	301	The bucket you are attempting to access must be addressed using the specified endpoint. Send all future requests to this endpoint.
PreconditionFailed	412	At least one of the preconditions you specified did not hold.
Redirect	307	Temporary redirect.
RestoreAlreadyInProgress	409	Object restore is already in progress.

NAME	CODE	DESCRIPTION
RequestIsNotMultiPartContent	400	Bucket POST must be of the enclosure-type multipart/form-data.
RequestTimeout	400	Your socket connection to the server was not read from or written to within the timeout period.
RequestTimeTooSkewed	403	The difference between the request time and the server's time is too large.
RequestTorrentOfBucketError	400	Requesting the torrent file of a bucket is not permitted.
ServiceUnavailable	503	Reduce your request rate.
SignatureDoesNotMatch	403	The request signature we calculated does not match the signature you provided. Check your secret access key and signing method.
SlowDown	503	Reduce your request rate
TemporaryRedirect	307	You are being redirected to the bucket while DNS updates.
TokenRefreshREquired	400	The provided token must be refreshed.
TooManyBuckets	400	You have attempted to create more buckets than allowed.
UnexpectedContent	400	This request does not support content.
UnresolvableGrantByEmailAddress	400	The email address you provided does not match any account on record.
UserKeyMustBeSpecified	400	The bucket POST must contain the specified field name. If it is specified, check the order of the fields.

Content Type

The request header must be set to the “application/json” content type. When using Python to create REST scripts, set the content-type header value to application/json.

Amazon S3 REST API Compatibility

Supported Bucket and Object Operations

BUCKET OPERATIONS	OBJECT OPERATIONS
DELETE Bucket	DELETE Object
GET Bucket (Version 1)	GET Object
GET Bucket (Version 2)	HEAD Object
GET Bucket location	Delete Multiple Objects
GET Bucket Object Versions	PUT Object
PUT Bucket	PUT Object Copy
List Multipart Uploads	Abort Multipart Upload
HEAD Bucket	Initiate Multipart Upload
GET Bucket Versioning	List Parts
PUT Bucket versioning	Upload Part
GET Bucket Lifecycle	Upload Part Copy
PUT Bucket Lifecycle	Complete Multipart Upload
DELETE Bucket Lifecycle	POST Object

Bucket Operations

Create, view, and manage buckets in FlashBlade object stores.

Common Request Headers

Most bucket operations use the following common request headers. See [AWS common request headers](#) for comprehensive information about the following request headers.

NAME	REQUIRED	DESCRIPTION
Authorization	Yes	Authorization string.
Content-Encoding	No	The applied content encodings to the object.
Content-Length	Yes	The size of the object in bytes.
Content-Type	No	The standard MIME type describing the format of the object data.
Content-MD5	No	The Base64-encoded 128-bit MD5 digest of the message without the headers.
Date	No	The current date and time per the requester.
Expect	No	The request uses 100-continue, the API waits for an acknowledgement before sending the request body. The request body is not sent if the request is rejected based on the headers.
Host	No	For path-style requests, the value is s3.amazonaws.com. For virtual-style requests, the value is bucketname.s23.amazonaws.com.
x-amz-content-sha256	No	The hash of the request payload when using signature version 4 for request authentication.
x-amz-date	No	The current date and time per the requester.

Common Response Headers

All bucket operations use the following common response headers. See [AWS common response headers](#) for comprehensive information about the following response headers.

NAME	DESCRIPTION
Content-Length	The size of the object in bytes.
Content-Type	The MIME type of the content.
Date	The S3 response date and time.
ETag	The entity tag (hash) of the object.
x-amz-version-id	The object version.

DELETE Bucket

Request: DELETE {bucket}

Removes an empty bucket from an object store.

Only empty buckets can be deleted. Use the GET Bucket request to check the contents of a bucket, and if necessary, delete its objects one-by-one prior to deleting the bucket.

Parameters

PARAMETER	TYPE	DESCRIPTION
None	n/a	n/a

Request Elements

This operation does not use request elements.

Response Elements

This operation does not return response elements.

GET Bucket (List Objects) Version 1

Request: GET {bucket}

Lists all objects in the specified bucket (maximum 1,000 objects in a single call), or, if no bucket is specified, lists all buckets in the object store.

Parameters

Refer to [AWS GET Bucket version 1 API](#) for comprehensive information about each parameter below.

PARAMETER	TYPE	DESCRIPTION
<code>delimiter</code>	String	The character used to group keys. Required: No
<code>Marker</code>	String	The key to start with when listing objects in a bucket. Required: No
<code>max-keys</code>	String	The maximum number of keys returned in the response body. Required: No
<code>prefix</code>	String	The specified prefix to which responses are limited. Required: No

Request Elements

This operation does not use request elements.

Response Elements

Refer to [AWS GET Bucket version 1 API](#) for comprehensive information about each response element below.

NAME	TYPE	DESCRIPTION
<code>Contents</code>	XML metadata	The metadata about each object returned.
<code>CommonPrefixes</code>	String	All keys rolled up in a common prefix
<code>Delimiter</code>	String	Rolls keys containing the same string between the prefix and delimiter into a single result element.
<code>DisplayName</code>	String	Object owner's name.
<code>Encoding-Type</code>	String	Encoding type used by Amazon S3 to encode object key names in the XML response.

ETag	String	The entity tag of the object.
ID	String	The object owner's ID.
IsTruncated	Boolean	Specifies whether (true or false) all results were returned.
Key	String	The object's key.
LastModified	Date	The date and time the object was last modified.
Marker	String	Where in the bucket that listing begins.
MaxKeys	String	The maximum number of keys returned in the response body.
Name	String	The bucket name.
NextMarker	String	The key name used as a marker in subsequent requests for the next set of objects.
Owner	String	The bucket's owner.
Prefix	String	Keys that begin with the indicated prefix.
Size	String	The object's size in bytes.
StorageClass	String	The storage class.

GET Bucket (List Object) Version 2

Request: GET {bucket}

Lists all objects in the specified bucket (maximum 1,000 objects in a single call), or, if no bucket is specified, lists all buckets in the object store.

Parameters

Refer to [AWS GET Bucket version 2 API](#) for comprehensive information about each parameter below.

PARAMETER	TYPE	DESCRIPTION
delimiter	String	The character used to group keys. Required: No
max-keys	String	The maximum number of keys returned in the response body. Required: No
prefix	String	The specified prefix to which responses are limited. Required: No

<code>start-after</code>	String	Returns key names after a specific object key in your key space. Required: No
--------------------------	--------	--

Request Elements

This operation does not use request elements.

Response Elements

Refer to [AWS GET Bucket version 2 API](#) for comprehensive information about each response element below.

NAME	TYPE	DESCRIPTION
<code>Contents</code>	XML metadata	The metadata about each object returned.
<code>CommonPrefixes</code>	String	All keys rolled up in a common prefix
<code>ContinuationToken</code>	String	Included in the response if sent with the request.
<code>Delimiter</code>	String	Rolls keys containing the same string between the prefix and delimiter into a single result element.
<code>DisplayName</code>	String	Object owner's name.
<code>Encoding-Type</code>	String	Encoding type used by Amazon S3 to encode object key names in the XML response.
<code>ETag</code>	String	The entity tag of the object.
<code>ID</code>	String	The object owner's ID.
<code>IsTruncated</code>	Boolean	Specifies whether (true or false) all results were returned.
<code>Key</code>	String	The object's key.
<code>KeyCount</code>	String	The number of keys included in the response.
<code>LastModified</code>	Date	The date and time the object was last modified.
<code>MaxKeys</code>	String	The maximum number of keys returned in the response body.
<code>Name</code>	String	The bucket name.
<code>NextContinuationToken</code>	String	Returned with a continuation token if the response is truncated.
<code>NextMarker</code>	String	The key name used as a marker in subsequent requests for the next set of objects.

Owner	String	The bucket's owner.
Prefix	String	Keys that begin with the indicated prefix.
Size	String	The object's size in bytes.
StartAfter	String	Included in the response if sent with the request.
StorageClass	String	The storage class.

GET Bucket Location

Request: GET {bucket}?location

Lists the bucket's region.

Parameters

PARAMETER	TYPE	DESCRIPTION
None	n/a	n/a

Request Elements

This operation does not use request elements.

Response Elements

Refer to [AWS GET Bucket location API](#) for comprehensive information about the response element below.

NAME	TYPE	DESCRIPTION
LocationConstraint	String	The region where the bucket resides. This implementation returns <code>on-prem</code> .

List Multipart Uploads

Request: GET {bucket}?uploads

Lists all in-progress multipart uploads.

Parameters

Refer to [AWS List multipart uploads API](#) for comprehensive information about each parameter below.

PARAMETER	TYPE	DESCRIPTION
<code>delimiter</code>	String	The character used to group keys. Required: No
<code>key-marker</code>	String	The key from which the multipart upload should begin. Required: No
<code>max-uploads</code>	String	The maximum number of multipart uploads returned in the response body. Required: No
<code>prefix</code>	String	The specified prefix to which responses are limited. Required: No
<code>upload-id-marker</code>	String	Used with the <code>key-marker</code> parameter to specify which listing begins the multipart upload. Required: No

Request Elements

This operation does not use request elements.

Response Elements

Refer to [AWS List multipart Uploads API](#) for comprehensive information about each response element below.

NAME	TYPE	DESCRIPTION
<code>Bucket</code>	XML metadata	The bucket name to which the multipart upload was initiated.
<code>CommonPrefixes</code>	String	If a delimiter was specified, the distinct key prefix containing the delimiter.
<code>CommonPrefixes.Prefix</code>	String	The substring of the key if the prefix parameter is not used.

Delimiter	String	The specified delimiter in the request.
DisplayName	String	The principal's name.
Encoding-Type	String	Encoding type used by Amazon S3 to encode object key names in the XML response.
ID	String	Provides the canonical user ID for AWS accounts or the user ARN value for IAM users.
Initiated	Date	The date and time at which the multipart upload was initiated.
Initiator	Container	The container element that identifies the initiator of the multipart upload.
IsTruncated	Boolean	Specifies whether (true or false) the returned list of multipart uploads is truncated.
Key	Integer	The key of the object.
KeyMarker	String	The key at or after which the listing began.
ListMultipartUploadsResult	Container	The response container.
ListMultipartUploadsResult.Prefix	String	The specified prefix when a prefix is provided in the request.
NextKeyMarker	String	The value to be used for the key marker request when a list is truncated.
NextUploadIdMarker	String	The value that should be used for the upload-id-marker when a list is truncated.
Owner	Container	The container element identifying the object owner.
StorageClass	String	The storage class.
Upload	Container	The container for elements related to a particular multipart upload.
UploadId	Integer	The multipart upload ID.
UploadIdMarker	String	The upload ID after which listing began.

GET Bucket Object Versions

Request: GET {bucket}?versions

Lists all metadata about the all versions of objects in the bucket.

Parameters

Refer to [AWS GET Bucket object versions API](#) for comprehensive information about each parameter below.

PARAMETER	TYPE	DESCRIPTION
delimiter	String	The character used to group keys. Required: No
key-marker	String	The key in the bucket from which you wish to begin listing. Required: No
max-keys	String	The maximum number of keys returned in the response body. Required: No
prefix	String	The specified prefix to which responses are limited. Required: No
version-id-marker	String	The object version from which you wish to begin listing. Required: No

Request Elements

This operation does not use request elements.

Response Elements

Refer to [AWS GET bucket object versions API](#) for comprehensive information about each response element below.

NAME	TYPE	DESCRIPTION
DeleteMarker	Container	The container for an object that is a delete marker.
DisplayName	String	The object owner's name.
Encoding-Type	String	Encoding type used by Amazon S3 to encode object key names in the XML response.
ETag	String	The entity tag of the object.

ID	String	The object owner's ID.
IsLatest	Boolean	Specifies whether (true or false) the object is the current version of the object.
IsTruncated	Boolean	Specifies whether (true or false) Amazon S3 returned all results satisfying the search criteria.
Key	String	The key of the object.
KeyMarker	String	The last key returned in a truncated response.
LastModified	Date	The date and time the object was last modified.
ListVersionsResult	Container	The container for the result.
MaxKeys	String	The maximum number of objects to return.
Name	String	The bucket owner's name.
NextKeyMarker	String	The first key not returned that satisfies the search criteria.
NextVersionIdMarker	String	The first object version not returned that satisfies the search criteria.
Owner	String	The bucket owner.
Prefix	String	Selects objects starting with the value as supplied by the prefix parameter.
Size	String	The object size in bytes.
StorageClass	String	Always STANDARD.
Version	Container	The version information container.
VersionId	String	The version ID of an object.
VersionIdMarker	String	The last version of the key returned in a truncated response.

HEAD Bucket

Request: HEAD {bucket}

Determines if the bucket exists and if you have access permission.

Parameters

PARAMETER	TYPE	DESCRIPTION
None	n/a	n/a

Request Elements

This operation does not use request elements.

Response Elements

This operation does not return response elements.

PUT Bucket

Request: PUT {bucket}

Creates a new, empty bucket that is owned by the user issuing the request.

Parameters

PARAMETER	TYPE	DESCRIPTION
None	n/a	n/a

Request Headers

PARAMETER	TYPE	DESCRIPTION
x-amz-acl	String	The canned ACL to apply to the bucket being created. Valid value is: private.

Request Elements

This operation does not use request elements.

Response Elements

This operation does not return response elements.

PUT Bucket Lifecycle

Request: PUT {bucket}?lifecycle

Creates a new lifecycle configuration for the bucket, or replaces an existing bucket lifecycle configuration.

Parameters

This operation does not use request parameters.

Request Headers

The following request header is supported.

NAME	REQUIRED	DESCRIPTION
Content-MD5	Yes	The Base64-encoded 128-bit MD5 digest of the message without the headers.

Request Elements

Refer to [AWS PUT Bucket lifecycle API](#) for comprehensive information about the XML request elements below.

NAME	TYPE	DESCRIPTION
LifecycleConfiguration	Container	Container for lifecycle rules. 1000 rules can be added. Required: Yes
NoncurrentDays	Positive integer	Specifies when noncurrent object versions expire. Required: Yes
NoncurrentVersionExpiration	Container	Specifies when noncurrent object versions expire. Upon expiration, S3 permanently deletes the noncurrent object versions. Required: Yes, if no action is present in the Rule.
Prefix	String	The object key prefix that identifies one or more objects to which the rule applies. A maximum of one prefix is allowed in a lifecycle rule filter. The maximum length allowed is 1024. Required: No
Rule	Container	The lifecycle rule container. It may contain up to 1000 rules. Required: Yes
Status	String	Can be enabled or disabled. If enabled, the rule is executed as scheduled. If disabled, the rule is ignored. Required: Yes

Response Headers

This operation uses only response headers that are common to most responses.

Response Elements

This operation does not return response elements.

DELETE Bucket Lifecycle

Request: DELETE {bucket}?lifecycle

Deletes the lifecycle configuration from a bucket.

Parameters

This operation does not use request parameters.

Request Headers

This operation uses only request headers that are common to most requests.

Request Elements

This operation does not use request elements.

Response Headers

This operation uses only response headers that are common to most responses.

Response Elements

This operation does not return response elements.

GET Bucket Lifecycle

Request: GET {bucket}?lifecycle

Returns the lifecycle configuration information set on a bucket.

Parameters

This operation does not use request parameters.

Request Headers

This operation uses only request headers that are common to most requests.

Request Elements

This operation does not use request elements.

Response Headers

This operation uses only response headers that are common to most responses.

Response Elements

Refer to [AWS GET Bucket lifecycle API](#) for comprehensive information about the response elements below.

NAME	TYPE	DESCRIPTION
LifecycleConfiguration	Container	Container for lifecycle rules. 1000 rules can be added.
NoncurrentDays	Positive integer	Specifies when noncurrent object versions expire.
NoncurrentVersionExpiration	Container	Specifies when noncurrent object versions expire. Upon expiration, S3 permanently deletes the noncurrent object versions.
Prefix	String	The object key prefix that identifies one or more objects to which the rule applies. A maximum of one prefix is allowed in a lifecycle rule filter. The maximum length allowed is 1024.
Rule	Container	The lifecycle rule container. It may contain up to 1000 rules.
Status	String	Can be enabled or disabled. If enabled, the rule is executed as scheduled. If disabled, the rule is ignored.

PUT Bucket Versioning

Request: PUT {bucket}?versioning

This operation sets a versioning state for a bucket. The version state can be set as Enabled or Suspended.

Parameters

PARAMETER	TYPE	DESCRIPTION
None	n/a	n/a

Request Headers

This operation does not use any request-specific headers.

Request Elements

Refer to [AWS PUT Bucket versioning API](#) for comprehensive information about the request elements below.

NAME	TYPE	DESCRIPTION
Status	Enum	Sets the buckets versioning state. Valid Values: Suspended Enabled
VersioningConfiguration	Container	Container for setting the versioning state.

Response Headers

This operation uses only response headers that are common to most responses.

Response Elements

This operation does not return response elements.

GET Bucket Versioning

Request: GET {bucket}?versioning

This operation gets the versioning state of a specified bucket. Each bucket version state can be Enabled or Suspended. If versioning has never been set on the bucket, the versioning state is NONE and VersioningConfiguration element is not returned in the response.

Request Parameters

This operation does not use request parameters.

Request Headers

This operation does not use any request-specific headers.

Request Elements

This operation does not use request elements.

Response Elements

Refer to [AWS GET bucket versioning API](#) for comprehensive information about the request elements below.

NAME	TYPE	DESCRIPTION
Status	Enum	The versioning state of the bucket. Valid Values: Suspended Enabled
VersioningConfiguration	Container	Container for the Status response element.

Object Operations

Create, view, and manage objects. In order to perform object operations, the bucket must already exist.

POST Object

Request: POST {bucket}

Adds an object to a specified bucket, using the `multipart/form-data` encoding.

Using POST with `x-www-form-urlencoded` bodies will be rejected with a 405 Method Not Allowed message.

AWS Signature Versions 2 and 4 are supported. Refer to [AWS Signature Version 2](#) and [AWS Signature Version 4](#) for comprehensive information about Signature Versions.

Request Parameters

This implementation of the operation does not use request parameters.

Form Fields

Refer to [AWS POST Object API](#) for comprehensive information about the form fields below.

PARAMETER	TYPE	DESCRIPTION
AWSAccessKeyID	String	The bucket owner can grant access to an anonymous user with the AWS access key ID. Required: Yes (V2)
acl	String	Specifies an Amazon S3 access control list. If an invalid access control list is specified, an error is generated. Required: No
Content-Type, Content-Encoding	String	The REST-specific headers. Required: No
file	File or text content	The file or text content. Required: Yes
key	String	The name of the uploaded key. Required: Yes

<code>policy</code>	String	The security policy permissions in the request. Required: Yes
<code>success_action_redirect,</code> <code>redirect</code>	String	The URL to which the client is redirected upon successful upload. NOTE: The <code>redirect</code> field name is deprecated. Required: No
<code>success_action_status</code>	String	The status code returned when the upload succeeds. Required: No

Response Headers

Refer to [AWS POST Object API](#) for comprehensive information about the response headers below.

PARAMETER	TYPE	DESCRIPTION
<code>success_action_redirect,</code> <code>redirect</code>	String	The URL to which the client is redirected on successful upload.
<code>x-amz-version-id</code>	String	The version of the object.

Response Elements

Refer to [AWS POST Object API](#) for comprehensive information about the response elements below.

PARAMETER	TYPE	DESCRIPTION
<code>Bucket</code>	String	The name of the bucket the object is stored in.
<code>ETag</code>	String	The entity tag.
<code>Key</code>	String	The object key name.
<code>Location</code>	String	The URI of the object.

DELETE Object

Request: DELETE {bucket}/{object}

Removes the specified object from the specified bucket and deletes the object in the background. Deletes only a single object.

Parameters

PARAMETER	TYPE	DESCRIPTION
None	n/a	n/a

Request Headers

This operation does not use any request-specific request headers.

Request Elements

This operation does not use request elements.

Response Headers

This table describes supported response headers for DELETE object requests.

NAME	DESCRIPTION
x-amz-version-id	The version of the delete marker created as a result of the DELETE operation. See AWS Delete Object API for comprehensive information about this response header.

Response Elements

This operation does not return response elements.

GET Object

Request: GET {bucket}/{object}

Retrieves the specified object from the specified bucket.

Parameters

Refer to [AWS GET Object API](#) for comprehensive information about the parameters below.

PARAMETER	TYPE	DESCRIPTION
response-content-encoding	String	Sets the Content-Encoding header of the response.
response-content-type	String	Sets the Content-Type header of the response.
versionId	String	Gets the version of the object specified by the given version ID.

Request Headers

This table describes supported request headers for `GET object` requests. See [AWS GET Object API](#) for comprehensive information about each request header below.

NAME	REQUIRED	DESCRIPTION
Range	No	Downloads the specified range of an object's bytes.
If-Match	No	Checks the entity tag (ETag) value before returning the object.
If-Modified-Since	No	Returns the object if it has been modified since the specified time.
If-None-Match	No	Checks the ETag value before returning the object.
If-Unmodified-Since	no	Returns the object if it has not been modified since the specified time.

Request Elements

This operation does not use request elements.

Response Headers

This table describes supported response headers for `GET object` requests.

NAME	DESCRIPTION
x-amz-version-id	The version of the object. See AWS GET Object API for comprehensive information about this response header.

Response Elements

This operation does not return response elements.

HEAD Object

Request: HEAD {bucket}/{object}

Returns metadata for the object without returning the object itself.

Parameters

PARAMETER	TYPE	DESCRIPTION
None	n/a	n/a

Request Headers

This operation does not use any request-specific request headers.

Request Elements

This operation does not use request elements.

Response Headers

This table describes supported response headers for HEAD object requests.

NAME	DESCRIPTION
Content-Length	The size of the object in bytes. See AWS common response headers for comprehensive information about this response header.
x-amz-version-id	The version of the object. See AWS Object API for comprehensive information about this response header.

Response Elements

This operation does not return response elements.

DELETE Multiple Objects

Request: POST {bucket}?delete

Removes multiple objects maximum of from the specified bucket—1000 objects can be deleted in a single call. This request has an XML document in the message body. Refer to [Request Elements in AWS delete multiple objects API](#) for comprehensive information.

Parameters

PARAMETER	TYPE	DESCRIPTION
delete	String	Specifies a delete operation.

Request Headers

See [AWS common request headers](#) for comprehensive information about the following request headers.

NAME	REQUIRED	DESCRIPTION
Authorization	No	Authorization string.
Content-Length	Yes	The size of the object in bytes.
Content-Type	No	The standard MIME type describing format of the object data.
Content-MD5	Yes	The Base64-encoded 128-bit MD5 digest of the message without the headers.
Date	No	The current date and time per the requester.
Expect	No	The request uses 100-continue, the API waits for an acknowledgement before sending the request body. The request body is not sent if the request is rejected based on the headers.
Host	No	For path-style requests, the value is s3.amazonaws.com. For virtual-style requests, the value is bucketname.s23.amazonaws.com.
x-amz-content-sha256	No	The hash of the request payload when using signature version 4 for request authentication.
x-amz-date	No	The current date and time per the requester.
x-amz-mfa	Conditional	The concatenation of the authentication device's serial number, a space, and the value displayed on the authentication device.

Request Elements

Refer to [AWS delete multiple objects API](#) for comprehensive information about the request elements below.

NAME	REQUIRED	TYPE	DESCRIPTION
Delete	Yes	Container	The container for the request.
Quiet	No	Boolean	The element to enable quiet mode for the request.
Object	Yes	Container	The container element that describes the object delete request.
Key	No	String	The key name of the object to delete.
VersionId	No	String	The version ID for the object to delete.

Response Headers

See [AWS common response headers](#) for comprehensive information about each response header below.

NAME	DESCRIPTION
Content-Length	The size of the object in bytes.
Content-Type	The MIME type of the content.
Date	The S3 response date and time.
ETag	The entity tag (hash) of the object.
x-amz-version-id	The object version.

Response Elements

Refer to [AWS delete multiple objects API](#) for comprehensive information about the response elements below.

NAME	TYPE	DESCRIPTION
Code	String	The status code for the result of the failed delete.
DefaultResult	Container	The response container.
Deleted	Container	The container element.
DeleteMarker	Boolean	Indicates whether (true or false) the request accessed a delete marker.

NAME	TYPE	DESCRIPTION
DeleteMarkerVersionId	String	The version of ID of the accessed delete marker.
Error	String	The container for a failed delete operation
Key	String	The key name for object that Amazon S3 attempted to delete.
Message	String	The error description.
VersionId	String	The version ID for the versioned object.

PUT Object

Request: PUT {bucket}/{object}

Adds an object to a bucket.

No partial objects

A success response means the object was added in its entirety. Partial objects are not added.

MD5 checks

When a request includes the MD5 header, the API returns an error if the received object's calculated MD5 value does not match the provided MD5 value. An ETag is returned with the object's MD5 value after the PUT.

Sending request headers

Use the 100-continue HTTP status code to send request headers before sending the request body. You can avoid sending the request body if request is rejected based on the headers (for instance, because of an authentication failure or redirect).

No support for object locking

When multiple PUT requests are received for the same object, the last request "wins" -- the content from other previous requests is overwritten. Object locking is not provided through the API.

Parameters

PARAMETER	TYPE	DESCRIPTION
None	n/a	n/a

Request Headers

This table describes supported `PUT object` request headers. Unless otherwise specified, see [AWS PUT Object API](#) for comprehensive information about each request header below.

NAME	REQUIRED	DESCRIPTION
Authorization	Yes	Authorization string. See AWS common request headers for comprehensive information about this request header.
Content-Length	Yes	The size of the object in bytes.
Content-MD5	No	The Base64-encoded 128-bit MD5 digest of the message without the headers.
Content-Type	No	The MIME type describing the format of the contents.
Expect	No	The request uses <code>100-continue</code> , the API waits for an acknowledgement before sending the request body. The request body is not sent if the request is rejected based on the headers.
Host	No	For path-style requests, the value is <code>s3.amazonaws.com</code> . For virtual-style requests, the value is <code>bucketname.s23.amazonaws.com</code> . See AWS common request headers for comprehensive information about this request header.
x-amz-acl	No	Sets the ACL of the object using the specified canned ACL. Valid value is: <code>private</code> .

Request Elements

This operation does not use request elements.

Response Headers

This table describes supported response headers for `PUT object` requests. See [AWS PUT Object API](#) for comprehensive information about each response header below.

NAME	DESCRIPTION
x-amz-version-id	The version of the object.

Response Elements

This operation does not return response elements.

PUT Object Copy

Request: PUT Object - Copy {bucket}/{object}

Retrieves and creates a copy of the specified object to the specified bucket.

Requirements

- The source bucket must be valid.
- READ access to the source object.

No support for storage classes

The current release does not support storage classes.

No support for user-defined metadata

The current release does not support user-defined metadata.

No support for object tags

The current release does not support object tagging.

No support for websites

The current release does not support buckets configured as websites.

Parameters

PARAMETER	TYPE	DESCRIPTION
None	n/a	n/a

Request Headers

This table describes supported request headers for PUT Object - Copy requests. See [AWS PUT Object - Copy API](#) for comprehensive information about each request header below.

NAME	REQUIRED	DESCRIPTION
x-amz-copy-source	Yes	The name of the source bucket and key name of the source object, separated by a slash (/).
x-amz-copy-source-if-match	No	Copies the object if its entity tag matches the specified tag.

NAME	REQUIRED	DESCRIPTION
x-amz-copy-source-if-none-match	No	Copies the object if its entity tag does not match the specified tag.
x-amz-copy-source-if-unmodified-since	No	Copies the object if it has not been modified since the specified time.
x-amz-copy-source-if-modified-since	No	Copies the object if it has been modified since the specified time.
x-amz-acl	No	The canned ACL to apply the object. Valid value is: private.

Request Elements

This operation does not use request elements.

Response Headers

This table describes supported response headers for `PUT Object - Copy` requests. See [AWS PUT Object - Copy API](#) for comprehensive information about each response header below.

NAME	DESCRIPTION
x-amz-source-version-id	The version of the source object.
x-amz-version-id	The version of the object.

Response Elements

Refer to [AWS PUT Object - Copy API](#) for comprehensive information about the response elements below.

NAME	TYPE	DESCRIPTION
CopyObjectResult	Container	The container for all response elements.

NAME	TYPE	DESCRIPTION
Etag	String	The Etag of the new object.
LastModified	String	The date the object was last modified.

Multipart Object Operations

Create, view, and manage the upload of large objects in smaller parts. In order to perform multipart object operations, the bucket must already exist.

Abort Multipart Upload

Request: `DELETE {bucket}/{object}?uploadId={uploadID}`

Aborts a multipart upload.

Parameters

PARAMETER	TYPE	DESCRIPTION
uploadID	String	The ID of the multipart upload.

Request Headers

See [AWS common request headers](#) for comprehensive information about the following request headers.

NAME	REQUIRED	DESCRIPTION
Authorization	Yes	Authorization string.
Content-Length	Yes	The size of the object in bytes.

NAME	REQUIRED	DESCRIPTION
Content-Type	No	The standard MIME type describing format of the object data.
Content-MD5	No	The Base64-encoded 128-bit MD5 digest of the message without the headers.
Date	No	The current date and time per the requester.
Expect	No	The request uses 100-continue, the API waits for an acknowledgement before sending the request body. The request body is not sent if the request is rejected based on the headers.
Host	No	For path-style requests, the value is s3.amazonaws.com. For virtual-style requests, the value is bucketname.s23.amazonaws.com.
x-amz-content-sha256	No	The hash of the request payload when using signature version 4 for request authentication.
x-amz-date	No	The current date and time per the requester.

Request Elements

This operation does not use request elements.

Response Headers

See [AWS common response headers](#) for comprehensive information about the following response headers.

NAME	DESCRIPTION
Content-Length	The size of the object in bytes.
Content-Type	The MIME type of the content.
Date	The S3 response date and time.
ETag	The entity tag (hash) of the object.
x-amz-version-id	The object version.

Response Elements

This operation does not return response elements.

List Parts

Request: GET {bucket}/{object}?uploadID={uploadID}

Lists the parts that have been uploaded for a specific multipart upload. This request includes an XML document in the message body of the response. Refer to [Response Elements in AWS list parts API](#) for comprehensive information.

Parameters

PARAMETER	TYPE	DESCRIPTION
uploadID	String	The ID of the multipart upload.

Request Headers

See [AWS common request headers](#) for comprehensive information about the following request headers.

NAME	REQUIRED	DESCRIPTION
Authorization	Yes	Authorization string.
Content-Length	Yes	The size of the object in bytes.
Content-Type	No	The standard MIME type describing format of the object data.
Content-MD5	No	The Base64-encoded 128-bit MD5 digest of the message without the headers.
Date	No	The current date and time per the requester.
Expect	No	The request uses 100-continue, the API waits for an acknowledgement before sending the request body. The request body is not sent if the request is rejected based on the headers.
Host	No	For path-style requests, the value is s3.amazonaws.com. For virtual-style requests, the value is bucketname.s23.amazonaws.com.
x-amz-content-sha256	No	The hash of the request payload when using signature version 4 for request authentication.
x-amz-date	No	The current date and time per the requester.

Request Elements

This operation does not use request elements.

Response Headers

See [AWS common response headers](#) for comprehensive information about each response header below.

NAME	DESCRIPTION
Content-Length	The size of the object in bytes.
Content-Type	The MIME type of the content.
Date	The S3 response date and time.
ETag	The entity tag (hash) of the object.
x-amz-version-id	The object version.

Response Elements

Refer to [AWS list parts API](#) for comprehensive information about the response elements below.

NAME	TYPE	DESCRIPTION
Bucket	String	The bucket name to which the upload was initiated.
DisplayName	String	The principal's name.
Encoding-Type	String	The encoding type used by S3 to encode object key names.
ETag	String	The returned entity tag when the part was uploaded.
ID	String	The canonical user ID if the principal is an AWS account, or the user ARN value if the principal is an IAM user.
Initiator	Container	The container element that identifies who initiated the upload.

NAME	TYPE	DESCRIPTION
IsTruncated	Boolean	Indicates whether a returned list is truncated.
Key	String	The object key.
LastModified	Date	The date and time the part was uploaded.
ListPartResult	Container	The response container.
MaxParts	Integer	The maximum number of parts allowed in the response.
NextPartNumberMarker	Integer	The last part in the list and value to use for the <code>part-number-marker</code> request parameter in a subsequent request, when a list is truncated.
Owner	Container	The container element that identifies the object owner.
Part	String	The container for elements related to a specific part.
PartNumber	Integer	The part number.
PartNumberMarker	Integer	The part number after which listing begins.
Size	Integer	The size of the uploaded part.
StorageClass	String	The storage class used to store the uploaded object.
UploadID	String	The upload ID.

NAME	TYPE	DESCRIPTION
x-amz-abort-date	String	The date at which the upload becomes eligible for abort operation if the bucket has a configured lifecycle rule with an action to abort incomplete multipart uploads.
x-amz-abort-rule-id	String	The applicable lifecycle configuration rule that defines the action to abort incomplete multipart uploads.

Initiate Multipart Upload

Request: POST {bucket}/{object}?uploads

Initiates a multipart upload and returns an upload ID.

Parameters

PARAMETER	TYPE	DESCRIPTION
None	n/a	n/a

Request Headers

See [AWS Initiate Multipart Upload API](#) for comprehensive information about the following request headers.

NAME	REQUIRED	DESCRIPTION
Cache-Control	No	Specifies caching behavior.
Content-Disposition	No	Specifies presentational information for the object.
Content-Encoding	No	The applied content encodings to the object.

NAME	REQUIRED	DESCRIPTION
Content-Type	No	The standard MIME type describing format of the object data.
Expires	No	The date and time at which the object is no longer cacheable for initiating multipart uploads.

Request Elements

This operation does not use request elements.

Response Headers

This operation does not use any response-specific response headers.

Response Elements

See [AWS Initiate Multipart Upload API](#) for comprehensive information about the following response elements.

NAME	TYPE	DESCRIPTION
InitiateMultipartUploadResult	Container	The response container.
Bucket	String	The bucket name to which the upload was initiated.
Key	String	The object key.
UploadId	String	The ID of the initiated upload.

Complete Multipart Upload

Request: POST {bucket}/{object}?uploadID={uploadID}

Completes a multipart upload by assembling previously uploaded parts.

Parameters

PARAMETER	TYPE	DESCRIPTION
uploadID	String	The ID of the multipart upload.

Request Headers

See [AWS common request headers](#) for comprehensive information about the following request headers.

NAME	REQUIRED	DESCRIPTION
Authorization	Yes	Authorization string.
Content-Length	Yes	The size of the object in bytes.
Content-Type	No	The standard MIME type describing format of the object data.
Content-MD5	No	The Base64-encoded 128-bit MD5 digest of the message without the headers.
Date	No	The current date and time per the requester.
Expect	No	The request uses 100-continue, the API waits for an acknowledgement before sending the request body. The request body is not sent if the request is rejected based on the headers.

NAME	REQUIRED	DESCRIPTION
Host	No	For path-style requests, the value is s3.amazonaws.com. For virtual-style requests, the value is bucketname.s23.amazonaws.com.
x-amz-content-sha256	No	The hash of the request payload when using signature version 4 for request authentication.
x-amz-date	No	The current date and time per the requester.

Request Elements

Refer to [AWS complete multipart upload API](#) for comprehensive information about the request elements below.

NAME	REQUIRED	TYPE	DESCRIPTION
CompleteMultipartUpload	Yes	Container	The container for the request.
Part	Yes	Container	The element container related to a previously uploaded part.
PartNumber	Yes	Integer	The part number.
ETag	Yes	String	The returned entity tag when the part was uploaded.

Response Headers

See [AWS Complete Multipart Upload API](#) for comprehensive information about the following response headers.

NAME	DESCRIPTION
x-amz-expiration	The expiration date and URL encoded rule ID.
x-amz-version-id	The version ID of the newly created object. Complete multipart upload.

Response Elements

See [AWS Complete Multipart Upload API](#) for comprehensive information about the following response elements.

NAME	TYPE	DESCRIPTION
CompleteMultipartUploadResult	Container	The response container.
Bucket	String	The bucket name that contains the newly created object.
ETag	String	The entity tag that identifies the newly created object's data.
Key	String	The object key.
Location	URI	The URI of the newly created object.

Upload Part

Request: PUT {bucket}/{object}?uploadID={uploadID}&part={partnumber}

Uploads a part in a multipart upload. A multipart upload must first be initiated before uploading any part.

Parameters

PARAMETER	TYPE	DESCRIPTION
partnumber	String	The part number of the multipart upload.
uploadID	Integer	The ID of the multipart upload.

Request Headers

Unless otherwise specified, see [AWS Upload Part API](#) for comprehensive information about each request header below.

NAME	REQUIRED	DESCRIPTION
Authorization	Yes	Authorization string. See AWS common request headers for comprehensive information about this request header.
Content-Length	Yes	The size of the object in bytes.
Content-MD5	No	The Base64-encoded 128-bit MD5 digest of the message without the headers.
Content-Type	No	The MIME type describing the format of the contents. See AWS common request headers for comprehensive information about this request header.

NAME	REQUIRED	DESCRIPTION
Date	No	The current date and time per the requester. See AWS common request headers for comprehensive information about this request header.
Expect	No	The request uses 100-continue, the API waits for an acknowledgement before sending the request body. The request body is not sent if the request is rejected based on the headers.
Host	No	For path-style requests, the value is s3.amazonaws.com. For virtual-style requests, the value is bucketname.s23.amazonaws.com. See AWS common request headers for comprehensive information about this request header.
x-amz-content-sha256	No	The hash of the request payload when using signature version 4 for request authentication. See AWS common request headers for comprehensive information about this request header.
x-amz-copy-source-if-match:{etag}	No	Perform a copy if the source object entity tag (ETag) matches the specified value.
x-amz-copy-source-if-none-match:{etag}	No	Perform a copy if the source object Etag differs from the specified value.
x-amz-copy-source-if-modified-since:{timestamp}	No	Perform a copy if the source object is modified after the specified time.
x-amz-copy-source-if-unmodified-since:{timestamp}	No	Perform a copy if the source object is not modified after the specified time.

NAME	REQUIRED	DESCRIPTION
x-amz-date	No	The current date and time per the requester. See AWS common request headers for comprehensive information about this request header.

Request Elements

This operation does not use request elements.

Response Headers

Unless otherwise specified, see [AWS Upload Part API](#) for comprehensive information about the following response headers.

NAME	DESCRIPTION
Content-Length	The size of the object in bytes. See AWS common response headers for comprehensive information about this response header.
Content-Type	The MIME type of the content. See AWS common response headers for comprehensive information about this response header.
Date	The S3 response date and time. See AWS common response headers for comprehensive information about this response header.
ETag	The entity tag (hash) of the object. See AWS common response headers for comprehensive information about this response header.
x-amz-version-id	The object version. See AWS common response headers for comprehensive information about this response header.

Response Elements

This operation does not return response elements.

Upload Part Copy

Request: PUT {bucket}/{object}?uploadID={uploadID}&part={partnumber}

Uploads a part in a multipart upload by copying data from an existing object.

Parameters

PARAMETER	TYPE	DESCRIPTION
partnumber	String	The part number of the multipart upload.
uploadID	Integer	The ID of the multipart upload.

Request Headers

Unless otherwise specified, see [AWS Upload Part - Copy API](#) for comprehensive information about each request header below.

NAME	REQUIRED	DESCRIPTION
Authorization	Yes	Authorization string. See AWS common request headers for comprehensive information about this request header.
Content-Length	Yes	The size of the object in bytes. See AWS common request headers for comprehensive information about this request header.
Content-MD5	No	The Base64-encoded 128-bit MD5 digest of the message without the headers. See AWS common request headers for comprehensive information about this request header.
Content-Type	No	The MIME type describing the format of the contents. See AWS common request headers for comprehensive information about this request header.

NAME	REQUIRED	DESCRIPTION
Date	No	The current date and time per the requester. See AWS common request headers for comprehensive information about this request header.
Expect	No	The request uses 100-continue, the API waits for an acknowledgement before sending the request body. The request body is not sent if the request is rejected based on the headers. See AWS common request headers for comprehensive information about this request header.
Host	No	For path-style requests, the value is s3.amazonaws.com. For virtual-style requests, the value is bucketname.s23.amazonaws.com. See AWS common request headers for comprehensive information about this request header.
x-amz-copy-source	Yes	The name of the source bucket and the source object key name.
x-amz-copy-source-range	No	The range of bytes to copy from the source bucket.
x-amz-content-sha256	No	The hash of the request payload when using signature version 4 for request authentication. See AWS common request headers for comprehensive information about this request header.
x-amz-date	No	The current date and time per the requester. See AWS common request headers for comprehensive information about this request header.

Request Elements

This operation does not use request elements.

Response Headers

Unless otherwise specified, see [AWS Upload Part - Copy API](#) for comprehensive information about each request header below.

NAME	DESCRIPTION
Content-Length	The size of the object in bytes. See AWS common response headers for comprehensive information about this response header.
Content-Type	The MIME type of the content. See AWS common response headers for comprehensive information about this response header.
Date	The S3 response date and time. See AWS common response headers for comprehensive information about this response header.
ETag	The entity tag (hash) of the object. See AWS common response headers for comprehensive information about this response header.
x-amz-copy-source-version-id	The version of the source object that was copied.
x-amz-version-id	The object version. See AWS common response headers for comprehensive information about this response header.

Response Elements

See [AWS Upload Part – Copy API](#) for comprehensive information about the following response elements.

NAME	TYPE	DESCRIPTION
CopyPartResult	Container	The response container.
ETag	String	The entity tag of the new part.

NAME	TYPE	DESCRIPTION
LastModified	String	The date that the part was last modified.