

FlashBlade Object Store REST API

Version 1.2

Copyright Statement

© 2020 Pure Storage, the Pure P Logo, and the marks on the Pure Trademark List at <https://www.purestorage.com/legal/productenduserinfo.html> are trademarks of Pure Storage, Inc. Other names are trademarks of their respective owners.

The Pure Storage products and programs described in this documentation are distributed under a license agreement restricting the use, copying, distribution, and decompilation/reverse engineering of the products. 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 NON-INFRINGEMENT, 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.

Pure Storage, Inc. 650 Castro Street Mountain View, CA 94041 <http://www.purestorage.com>

Direct comments to DocumentFeedback@purestorage.com.

Table of Contents

Chapter 1 Introduction.....	5
Chapter 2 REST API Overview.....	6
What is REST?.....	6
FlashBlade Object Store REST API.....	7
Authentication.....	7
Supported Protocols.....	8
Pre-signed URLs.....	8
SDK Best Practice.....	8
Response Codes.....	8
Chapter 3 Amazon S3 REST API Compatibility.....	14
Supported Bucket and Object Operations.....	14
Bucket Operations.....	15
Common Request Headers.....	15
Common Response Headers.....	16
DeleteBucket.....	16
ListBuckets (List Objects) Version 1.....	17
ListObjectsV2.....	19
GetBucketLocation.....	21
ListMultipartUploads.....	22
ListBucket Object Versions.....	24
HeadBucket.....	26
PutBucket.....	27
PutBucketLifecycleConfiguration.....	28
DeleteBucketLifecycle.....	29
GetBucketLifecycleConfiguration.....	30
PutBucketVersioning.....	31
ListBucket Versioning.....	32
Object Operations.....	33
PostObject.....	33
DeleteObject.....	36
GetObject.....	37
HeadObject.....	38
DeleteMultipleObjects.....	40

PutObject.....	43
PutObject Copy.....	46
Abort Multipart Upload.....	49
ListParts.....	51
InitiateMultipartUpload.....	53
CompleteMultipartUpload.....	55
UploadPart.....	57
UploadPartCopy.....	61

Chapter 1

Introduction

This guide describes how to access 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.

This guide describes the AWS S3 API requests, responses, and parameters supported by Pure Storage. Links contained in this document to AWS documentation are for informational purposes only. Additional parameters shown in the AWS documentation that are not listed here are not supported by Pure Storage.

This 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.

Chapter 2

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:

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

```
bucket
bucket/object
```

Table 2.1 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

- 2 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.

FlashBlade 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](#).

SDK Best Practice

If using an SDK, it is recommended to use operations that directly copy objects as-is, such as boto3's `client.copy_object`, rather than copy helpers such as awscli's `s3 cp` command and boto3's `client.copy` function. The former options will take advantage of FlashBlade's S3 Fast Copy feature, which was introduced in Purity//FB 3.0.

Response Codes

Table 2.2 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.

Name	Code	Description
Bucket AlreadyExists	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.
Bucket NotEmpty	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.
InternalError	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.
InvalidEncryptionAlgorithmError	400	The encryption request is invalid. The valid value is AES256.
InvalidLocaionConstraint	400	The specified location restraint is invalid.

Name	Code	Description
InvalidObject State	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.

Name	Code	Description
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.
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.

Name	Code	Description
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.
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.

Name	Code	Description
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.

Chapter 3

Amazon S3 REST API Compatibility

Supported Bucket and Object Operations

Bucket Operations	Object Operations
DeleteBucket	DeleteObject
ListBucket (Version 1)	GetObject
ListObjectV2 (Version 2)	HeadObject
GetBucketLocation	DeleteMultipleObjects
ListBucket Object Versions	PutObject
PutBucket	PutObject Copy
ListMultipartUploads	AbortMultipartUpload
HeadBucket	InitiateMultipartUpload
ListBucket Versioning	ListParts
PutBucketVersioning	UploadPart
GetBucketLifecycle	UploadPartCopy
PutBucketLifecycle	CompleteMultipartUpload
DeleteBucketLifecycle	PostObject

Bucket Operations

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

Common Request Headers

Most bucket and object operations use the following common request and response 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	No	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 body without the headers.
Date	No	The current date and time per the requester.
Expect	No	When 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, the value will be data_vip of the FlashBlade. For virtual-style addressing, the value will be bucketname.s3.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 and object 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.

DeleteBucket

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.

ListBuckets (List Objects) Version 1

 **Note:** It is recommended to use ListObjectV2.

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. It is recommended to use Version 2.

Parameters

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

Parameter	Type	Description
delimiter	String	The character used to group keys. Required: No
encoding-type	String	Encoding type used by Amazon S3 to encode object keys in the response.
Marker	String	The key to start with when listing objects in a bucket. 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

Request Elements

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

This operation does not use request elements.

Request Element	Type	Description
Contents	XML metadata	The metadata about each object returned.
CommonPrefixes	String	All keys rolled up in a common prefix
Delimiter	String	Rolls keys containing the same string between the prefix and delimiter into a single result element.
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.
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.
Prefix	String	Keys that begin with the indicated prefix.
Size	String	The object's size in bytes.

Request Element	Type	Description
StorageClass	String	The storage class.

ListObjectsV2

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
continuation-token	String	Included in the response if sent with the request.
delimiter	String	The character used to group keys. Required: No
encoding-type	String	Encoding type used by Amazon S3 to encode object keys in the response.
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
start-after	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.

Response Element	Type	Description
Contents	XML metadata	The metadata about each object returned.
CommonPrefixes	String	All keys rolled up in a common prefix
ContinuationToken	String	Included in the response if sent with the request.
Delimiter	String	Rolls keys containing the same string between the prefix and delimiter into a single result element.
Encoding-Type	String	Encoding type used by Amazon S3 to encode object key names in the XML response.
ETa	String	The entity tag of the object.
IsTruncated	Boolean	Specifies whether (true or false) all results were returned.
Key	String	The object's key.
KeyCount	String	The number of keys included in the response.
LastModified	Date	The date and time the object was last modified.
MaxKeys	String	The maximum number of keys returned in the response body.
Name	String	The bucket name.
NextContinuationToken	String	Returned with a continuation token if the response is truncated.

Response Element	Type	Description
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.

GetBucketLocation

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.

Response Element	Type	Description
LocationConstraint	String	The region where the bucket resides. This implementation returns on-prem .

ListMultipartUploads

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
delimiter	String	The character used to group keys. Required: No
encoding-type	String	Encoding type used by Amazon S3 to encode object keys in the response.
key-marker	String	The key from which the multipart upload should begin. Required: No
max-uploads	String	The maximum number of multipart uploads returned in the response body. Required: No
prefix	String	The specified prefix to which responses are limited. Required: No
upload-id-marker	String	Used with the key-marker 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.

Response Element	Type	Description
Bucket	XML metadata	The bucket name to which the multipart upload was initiated.
CommonPrefixes	String	If a delimiter was specified, the distinct key prefix containing the delimiter.
CommonPrefixes.Prefix	String	The substring of the key if the prefix parameter is not used.
Delimiter	String	The specified delimiter in the request.
Encoding-Type	String	Encoding type used by Amazon S3 to encode object key names in the XML response.
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.

Response Element	Type	Description
NextUploadIdMarker	String	The value that should be used for the upload-id-marker when a list is truncated.
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.

ListBucket 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
encoding-type	String	Encoding type used by Amazon S3 to encode object keys in the response.
key-marker	String	The key in the bucket from which you wish to begin listing. Required: No

Parameter	Type	Description
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.

Response Element	Type	Description
DeleteMarker	Container	The container for an object that is a delete marker.
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.
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.

Response Element	Type	Description
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.
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.

HeadBucket

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.

PutBucket

Request: PUT {bucket}

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

Parameters

This operation does not use request parameters.

Request Headers

Request Header	Type	Description
x-amz-acl	String	Sets the ACL of the bucket being created using the specified canned ACL. This header cannot be blank. Valid Value: private .

Request Elements

This operation does not use request elements.

Response Elements

This operation does not return response elements.

PutBucketLifecycleConfiguration

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.

Request Header	Required	Description
Content-MD5	Yes	The Base64-encoded 128-bit MD5 digest of the message body or message payload without the headers.
Host	No	For path-style, the value will be data_vip of the FlashBlade. For virtual-style addressing, the value will be bucketname.s3.amazonaws.com.

Request Elements

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

Request Element	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

Request Element	Type	Description
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.

DeleteBucketLifecycle

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.

GetBucketLifecycleConfiguration

Request: PUT {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 PUT Bucket lifecycle API](#) for comprehensive information about the response elements below.

Response Element	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.

PutBucketVersioning

Request: PUT {bucket}?versioning

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

Parameters

This operation does not use request parameters.

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.

Request Element	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.

ListBucket 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.

Response Element	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.

PostObject

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

Parameter	Type	Description
x-amz-meta-*	String	The user-defined metadata for the current object. It should follow with a key value pair. Multiple pairs are allowed. Example: • x-amz-meta-key1: value1 • x-amz-meta-key2: value2 Required: No

Response Headers

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

Response Header	Type	Description
success_action_redirect, redirect	String	The URL to which the client is redirected on successful upload.
x-amz-version-id	String	The version of the object.

Response Elements

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

Response Element	Type	Description
Bucket	String	The name of the bucket the object is stored in.
ETag	String	The entity tag.
Key	String	The object key name.
Location	String	The URI of the object.

DeleteObject

Request: DELETE {bucket}/{object}

Removes the specified object from the specified bucket and deletes the object in the background. Deletes only a single object along with its attached custom metadata.

Parameters

Parameter	Type	Description
versionId	String	Version ID used to reference a specific version of a project.

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.

Request Element	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.

GetObject

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
<code>response-content-encoding</code>	String	Sets the Content-Encoding header of the response.
<code>response-content-type</code>	String	Sets the Content-Type header of the response.
<code>versionId</code>	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.

Request Header	Required	Description
<code>Range</code>	No	Downloads the specified range of an object's bytes.
<code>If-Match</code>	No	Checks the entity tag (ETag) value before returning the object.
<code>Host</code>	No	For path-style, the value will be <code>data_vip</code> of the FlashBlade. For virtual-style addressing, the value will be <code>bucketname.s3.amazonaws.com</code> .
<code>If-Modified-Since</code>	No	Returns the object if it has been modified since the specified time.
<code>If-None-Match</code>	No	Checks the ETag value before returning the object.

Request Header	Required	Description
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.

Request Element	Description
Content-Range	The portion of the object returned in the response.
Last-Modified	The last modified date of the object.
x-amz-version-id	The version of the object. See AWS GET Object API for information about this response header.
x-amz-meta-*	The value/values for each custom metadata key. Example: • x-amz-meta-key1: val1-1,val1-2,val1-3 • x-amz-meta-key2: val2

Response Elements

This operation does not return response elements.

HeadObject

Request: HEAD {bucket}/{object}

Returns metadata for the object without returning the object itself.

Parameters

Parameter	Type	Description
If-Match	String	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	String	Return the object only if it has been modified since the specified time, otherwise return a 304 (not modified).
If-None-Match	String	Return the object only if its entity tag (ETag) is different from the one specified, otherwise return a 304 (not modified).
If-Unmodified-Since	String	Return the object only if it has not been modified since the specified time, otherwise return a 412 (precondition failed).
versionId	String	VersionId used to reference a specific version of the object.

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.

Response Header	Description
Content-Length	The size of the object in bytes. See AWS common response headers for comprehensive information about this response header.
Last-Modified	The last modified date of the object.

Response Header	Description
x-amz-delete-marker	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-version-id	The version of the object. See AWS Object API for comprehensive information about this response header.
x-amz-meta-*	The value/values for each custom metadata key. Example: • x-amz-meta-key1: val1-1,val1-2,val1-3 • x-amz-meta-key2: val2

Response Elements

This operation does not return response elements.

DeleteMultipleObjects

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.

Request Header	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, the value will be data_vip of the FlashBlade. For virtual-style addressing, the value will be bucketname.s3.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 delete multiple objects API](#) for comprehensive information about the request elements below.

Request Element	Type	Description
Delete	Container	The container for the request. Required: Yes
Quiet	Boolean	The element to enable quiet mode for the request. Required: No

Request Element	Type	Description
Object	Container	The container element that describes the object delete request. Required: Yes
Key	String	The key name of the object to delete. Required: No
VersionId	String	The version ID for the object to delete. Required: No

Response Headers

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

Response Header	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.

Response Element	Type	Description
Code	String	The status code for the result of the failed delete.

Response Element	Type	Description
DefaultResult	Container	The response container.
Deleted	Container	The container element.
DeleteMarker	Boolean	Indicates whether (true or false) the request accessed a delete marker.
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.

PutObject

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.

Support for User-defined Metadata

The user can perform a PUT operation with one or more headers in the form of `x-amz-meta-*` header, specifying the user-defined metadata of this object. The file size limit of the user-defined metadata is 2KB. Metadata size is calculated by taking the sum of the number of bytes of each key and value with the UTF-8 encoding.

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.

Request Header	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.

Request Header	Required	Description
Content-Type	No	The MIME type describing the format of the contents.
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, the value will be data_vip of the FlashBlade. For virtual-style addressing, the value will be bucketname.s3.amazonaws.com . See AWS common request headers for comprehensive information about this request header.
Metadata	No	The user-defined metadata that the user wants to include in the object.
x-amz-acl	Yes	Sets the ACL of the object using the specified canned ACL. This header cannot be blank. Valid Value: private .
x-amz-meta-*	No	The user-defined metadata for the current object. It should follow with a key value pair. Multiple pairs are allowed. Example: • x-amz-meta-key1: value1-1 • x-amz-meta-key1: value1-2 • x-amz-meta-key2: value2

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.

Response Header	Description
ETag	The entity tag (hash) of the object.
x-amz-version-id	The version of the object.

Response Elements

This operation does not return response elements.

PutObject 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.

Support for User-defined Metadata

When performing a Copy operation for an object, there are two options for user-defined metadata attached to this object.

COPY(default option): directly copy the the user-defined metadata from the original source object.

REPLACE: user provides a new user-defined metadata in the request, and replace metadata only when copying the original source object.

"MetadataDirective" specifies whether the user wants to perform a "COPY" or "REPLACE".

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.

Request Header	Required	Description
Host	No	For path-style, the value will be data_vip of the FlashBlade. For virtual-style addressing, the value will be bucketname.s3.amazonaws.com.
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.
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.

Request Header	Required	Description
x-amz-acl	Yes	Sets the ACL to be applied to the object using the specified canned ACL. This header cannot be blank. Valid Value: private .
x-amz-metadata-directive	No	The user-defined metadata directive used when the user wants to put-copy the current object. Its value could be REPLACE or COPY. If not provided, then default value is COPY.
x-amz-meta-*	No	The user-defined metadata for the current object. It should follow with a key value pair. Multiple pairs are allowed. Example: • x-amz-meta-key1: value1-1 • x-amz-meta-key1: value1-2 • x-amz-meta-key2: value2

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.

Response Header	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.

Response Element	Type	Description
CopyObjectResult	Container	The container for all response elements.
ETag	String	The entity tag (hash) 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.

Request Header	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.

Request Header	Required	Description
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, the value will be data_vip of the FlashBlade. For virtual-style addressing, the value will be bucketname.s3.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.

Response Header	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.

ListParts

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
max-parts	String	Sets the maximum number of parts to return in the response. Default: 1000
part-number-marker	String	Specifies the part after which listing should begin. Only parts with higher part numbers will be listed.
uploadID	String	The ID of the multipart upload.

Request Headers

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

Request Header	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.

Request Header	Required	Description
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, the value will be data_vip of the FlashBlade. For virtual-style addressing, the value will be bucketname.s3.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.

Response Header	Description
Date	The S3 response date and time.

Response Elements

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

Response Element	Type	Description
Bucket	String	The bucket name to which the upload was initiated.
Encoding-Type	String	The encoding type used by S3 to encode object key names.

Response Element	Type	Description
ETag	String	The returned entity tag when the part was uploaded.
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 part-number-marker request parameter in a subsequent request, when a list is truncated.
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.

InitiateMultipartUpload

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

Initiates a multipart upload and returns an upload ID.

Support for User-defined Metadata

The user can perform a PUT operation for an object with a "Metadata" header, specifying the user-defined metadata of this object.

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.

Request Header	Required	Description
Content-Encoding	No	The applied content encodings to the object.
Host	No	For path-style, the value will be data_vip of the FlashBlade. For virtual-style addressing, the value will be bucketname.s3.amazonaws.com .
x-amz-meta-*	No	The user-defined metadata for the current object. It should follow with a key value pair. Multiple pairs are allowed. Example: • x-amz-meta-key1: value1-1 • x-amz-meta-key1: value1-2 • x-amz-meta-key2: value2

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.

Response Element	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.

CompleteMultipartUpload

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.

Request Header	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.

Request Header	Required	Description
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, the value will be data_vip of the FlashBlade. For virtual-style addressing, the value will be bucketname.s3.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.

Request Element	Type	Description
CompleteMultipartUpload	Container	The container for the request. Required: Yes
Part	Container	The element container related to a previously uploaded part. Required: Yes
PartNumber	Integer	The part number. Required: Yes

Request Element	Type	Description
ETag	String	The returned entity tag (hash) when the part was uploaded. Required: Yes

Response Headers

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

Response Header	Description
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.

Response Element	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.

UploadPart

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.

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

Request Header	Description
Date	The current date and time per the requester. See AWS common request headers for comprehensive information about this request header. Required: No
Expect	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. Required: No
Host	For path-style, the value will be data_vip of the FlashBlade. For virtual-style addressing, the value will be bucketname.s3.amazonaws.com. See AWS common request headers for comprehensive information about this request header. Required: No
x-amz-content-sha256	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. Required: No
x-amz-copy-source-if-match:{etag}	Perform a copy if the source object entity tag (ETag) matches the specified value. Required: No
x-amz-copy-source-if-none-match:{etag}	Perform a copy if the source object ETag differs from the specified value. Required: No

Request Header	Description
<code>x-amz-copy-source-if-modified-since:</code> , <code>{timestamp}</code>	Perform a copy if the source object is modified after the specified time. Required: No
<code>x-amz-copy-source-if-unmodified-since:</code> , <code>{timestamp}</code>	Perform a copy if the source object is not modified after the specified time. Required: No
<code>x-amz-date</code>	The current date and time per the requester. See AWS common request headers for comprehensive information about this request header. Required: No

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.

Response Header	Description
<code>Content-Length</code>	The size of the object in bytes. See AWS common response headers for comprehensive information about this response header.
<code>Content-Type</code>	The MIME type of the content. See AWS common response headers for comprehensive information about this response header.
<code>Date</code>	The S3 response date and time. See AWS common response headers for comprehensive information about this response header.

Response Header	Description
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.

UploadPartCopy

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.

Request Header	Required	Description
Authorization	Yes	Authorization string. See AWS common request headers for comprehensive information about this request header.

Request Header	Required	Description
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.
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, the value will be data_vip of the FlashBlade. For virtual-style addressing, the value will be bucketname.s3.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.

Request Header	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 - Copy API](#) for comprehensive information about each request header below.

Response Header	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.

Response Element	Type	Description
CopyPartResult	Container	The response container.
ETag	String	The entity tag (hash) of the new part.
LastModified	String	The date that the part was last modified.



Pure Storage, Inc.

Twitter: @purestorage

650 Castro Street, Suite 400

Mountain View, CA 94041

T: 650-290-6088

F: 650-625-9667

Sales: sales@purestorage.com

Support: support@purestorage.com

Media: pr@purestorage.com

General: info@purestorage.com