Wasabi API Guide

wasabi
hot cloud storage

Copyright

Notice
Information in this guide is subject to change without notice. Companies, names, and data used in examples herein are fictitious unless otherwise noted. No part of this guide may be reproduced or transmitted in any form by means electronic or mechanical, for any purpose, without express written permission of Wasabi Technologies, Inc.

Trademarks
WASABI and the WASABI Logo are trademarks of Wasabi Technologies, Inc. and may not be used without permission of Wasabi Technologies, Inc. All other names are used for identification purposes only and are trademarks or registered trademarks of their respective companies.

Contents

Preface

1   Introduction

2   REST API Introduction

3   Operations on Buckets

4   Operations on Objects

5   IAM and STS Support

6   Compatible Libraries and Applications

Preface

About This Manual
This Guide is intended to describe variations in functionality as they pertain to using the Wasabi API relative to Amazon Web Services' Simple Storage Service (AWS S3) and Identity and Access Management (AWS IAM) APIs. Wasabi's object storage service is built to be 100% bit-compatible with the AWS S3 and IAM APIs. In certain cases, Wasabi provides additional functionality relative to AWS S3 (or implements functionality in a different manner than AWS S3).

Related Documentation
Additional Wasabi documentation includes:

◆ *Wasabi Getting Started*—Provides a quick overview of basic Wasabi features with procedures for new and prospective users. This does not replace the *Wasabi Management Console User Guide*.

◆ *Wasabi Management Console User Guide*—Provides details about Wasabi Management Console features with procedures for new and experienced users. It introduces you to basic Wasabi features to sign up to use Wasabi, sign into Wasabi, create a bucket, store objects (folders and files) in the bucket, create a user account, create groups, create policies, create roles, and add an access key to an account.

Conventions   This document uses the following typographical conventions:

Convention	Description
IMPORTANT	Stresses the importance of the information presented.
NOTE	Calls attention to additional information or an explanation.
Bold Text	Used to indicate specific text that you need to enter (key in) or a button that you need to click.
Italics Text	Used to indicate titles of books or sections within this document, and for general emphasis.

Technical Support   The Wasabi web site has the latest product information. Use this resource to review product information, download software updates, contact technical support, access the Wasabi knowledge base, or provide feedback on Wasabi documentation and online help:

www.wasabi.com

Use this email address to contact a Wasabi Customer Support representative:

support@wasabi.com

1 Introduction

Wasabi is a cloud-based object storage service for a broad range of applications and use cases. Wasabi is designed for individuals and organizations that require a high-performance, reliable, and secure data storage infrastructure at minimal cost.

Wasabi's object storage service is built to be 100% bit-compatible with Amazon Web Services' Simple Storage Service (AWS S3) and Identity and Access Management (AWS IAM) APIs. This means that any existing AWS S3-compatible application or gateway device will work seamlessly with Wasabi.

In certain cases, Wasabi provides additional functionality relative to AWS S3 (or implements functionality in a different manner than AWS S3). The primary purpose of this guide is to describe these variations in functionality as they pertain to using the Wasabi API relative to the AWS S3 and IAM APIs.

Given that Wasabi is built to be 100% bit-compatible with AWS S3, the AWS documentation for S3 and IAM is a complementary resource to the information provided in this guide. The following AWS document apply to the Wasabi API design:

◆ AWS S3 capabilities referenced in this Guide are based on API version 2006-03-01, described in the Amazon Simple Storage Service API Reference:
http://docs.aws.amazon.com/AmazonS3/latest/API/s3-api.pdf

◆ AWS IAM capabilities referenced in this Guide are based on API version 2010-05-08, described in the AWS Identity and Access Management API Reference: http://docs.aws.amazon.com/IAM/latest/APIReference/iam-api.pdf

Throughout this guide, for the sake of brevity, the Wasabi API functionality is designated as "Wasabi" and the AWS S3 API functionality is designed as "AWS S3."

In addition, examples indicate access to buckets created in the East Coast (us-east-1) region. Note, however, that you must use URLs based on the region to access buckets. If the bucket was created in the East Coast (us-east-1) region, use:

s3.wasabisys.com

If the bucket was created in the West Coast (us-west-1) region, use:

s3.us-west-1.wasabisys.com

Using the incorrect URL will enable you to do a GET, but not a PUT or DELETE function.

2

REST API Introduction

This section includes information for:

Host Designation Wasabi supports both the path-style and virtual-style requests available with AWS S3. A different host name is required (for example, s3.wasabisys.com versus s3.amazonaws.com). Wasabi recommends using path-style requests as shown in all examples in this Guide (for example, http://s3.wasabisys.com/my-bucket/my-object) because the path-style offers the greatest flexibility in bucket names, avoiding domain name issues.

Error Responses Where possible, Wasabi utilizes the exact error responses given by AWS S3. Some additional error codes are provided for additional functionality in Wasabi.

HTTP Error Codes

Error Code	Wasabi Description	HTTP Status Code
Unavailable	Temporarily unavailable	503
Unrecoverable	DataUnrecoverable	
NotImplemented	Not Implemented	501
DBEntityNotfound	No such entity	404
DBEntityFound	EntityAlreadyExists	409
DisconnectError	ConnectionClosed	410
StatusInternalServerError	InternalError, We encountered an internal error. Please contact customer support.	500
StatusForbidden	AccessDenied	403
	AccessForbidden	
	AccountNotActivated	
	AccountProblem	

Error Code	Wasabi Description	HTTP Status Code
	AuthenticationLockout, There have been too many attempts to access the account with the wrong credentials. Please try again in five minutes.	
	InvalidAccessKeyId, The AWS Access Key Id you provided does not exist in our records.	
	InvalidAuthenticationCode, Authentication code for device is not valid.	
	RequestTimeTooSkewed, The difference between the request time and the current time is too large.	
	InvalidObjectState, The operation is not valid for the current state of the object.	
StatusBadRequest	AuthorizationHeaderMalformed	400
	AuthorizationQueryParametersError	
	BadDigest, The Content-MD5 you specified did not match what we received.	
	BadRequest, An error occurred when parsing the HTTP request.	
	IncompleteSignature, Request must contain a signature that conforms to AWS standards.	
	InvalidAction	
	InvalidArgument	
	InvalidDigest, The Content-MD5 you specified was invalid.	
	InvalidInput	

Error Code	Wasabi Description	HTTP Status Code
	InvalidParameterValue, An invalid or out-of-range value was supplied for the input parameter.	
	InvalidRequest	
	MalformedPolicy	
	MalformedPolicyDocument	
	PasswordPolicyViolation	
	ValidationError	
	XAmzContentSHA256Mismatch, The provided 'x-amz-content-sha256' header does not match what was computed.	
	EncryptionNotAllowed, User provided encryption keys are not allowed in this operation.	
	The object was stored using a form of Server Side Encryption. The correct parameters must be provided to retrieve the object.	
	EntityTooLarge, Your proposed upload is larger than the maximum allowed size.	
	EntityTooSmall, Your proposed upload is smaller than the minimum allowed size.	
	IllegalComplianceRequest	
	IllegalVersioningConfigurationException, The versioning configuration specified in the request is not valid.	
	IncompleteBody, u did not provide the number of bytes specified by the Content-Length HTTP header.	

Error Code	Wasabi Description	HTTP Status Code
	InvalidACLRequest, You must provide only one of either ACL headers or an XML body when setting ACLs.	
	InvalidArgument	
	InvalidBucketName, The specified bucket is not valid.	
	InvalidEncryptionAlgorithmError, The Encryption request you specified is not valid. Supported value: AES256.	
	InvalidPartOrder, The list of parts was not in ascending order. Parts must be ordered by part number.	
	InvalidPart, One or more of the specified parts could not be found. The part may not have been uploaded, or the specified entity tag may not match the part's entity tag.	
	InvalidPolicyDocument, The content of the form does not meet the conditions specified in the policy document.	
	KeyTooLongError, Your key is too long.	
	LoggingBucketSameOwner, The logging target bucket must have the same owner as the bucket being logged.	
	MalformedXML, The XML you provided was not well formed or did not validate against our published schema.	
	MetadataTooLarge, Your metadata headers exceed the maximum allowed metadata size.	
	RequestTimeout, Your socket connection to the server was not read from or written to within the timeout period.	
	TooManyBuckets, You have attempted to create more buckets than allowed.	

Error Code	Wasabi Description	HTTP Status Code
	TooManyComponents, A composite object may not have more than 1024 components.	
	UnresolvableGrantByEmailAddress, The email address you provided does not match any account on record.	
	StorageQuotaExceeded, Your account has surpassed it's storage limit.	
	BucketAccountInactive, The account that owns this bucket is not active.	
StatusGone	ConnectionClosed, Network connection was closed.	410
StatusUnprocessableEntity	DataUnrecoverable, The data in the request is unrecoverable. Please contact customer support.	422
StatusConflict	DeleteConflict	409
	EntityAlreadyExists	
	EntityTemporarilyUnmodifiable, The entity is temporarily unmodifiable. Please try again later.	
	Limit Exceeded	
	OperationAborted, A conflicting conditional operation is currently in progress against this resource. Please try again.	
	BucketAlreadyExists	
	BucketNotEmpty	
	ComplianceSettingsLocked, The compliance settings are now locked and cannot be changed.	
StatusNotFound	NoSuchEntity	404

Error Code	Wasabi Description	HTTP Status Code
	NoSuchLifecycleConfiguration, The lifecycle configuration does not exist.	
	NoSuchBucket, The specified bucket does not exist.	
	NoSuchBucketPolicy, The bucket policy does not exist.	
	NoSuchKey, The specified key does not exist.	
	NoSuchReplicationConfiguration, The replication configuration does not exist.	
	NoSuchTagSetError, There is no tag set associated with the bucket.	
	NoSuchUpload, The specified upload does not exist. The upload ID may be invalid, or the upload may have been aborted or completed.	
	NoSuchVersion, The specified version does not exist.	
StatusNotImplemented	NotImplemented, A header you provided implies functionality that is not implemented.	501
StatusServiceUnavailable	TemporarilyUnavailable, Resources for this operation are temporarily unavailable. Please try again later.	503
StatusFound	KeyAlreadyExists	302
StatusMethodNotAllowed	MethodNotAllowed, This method is not allowed for other than the account owner.	405
	MethodNotAllowed, The specified method is not allowed against this resource.	
StatusLengthRequired	MissingContentLength, You must provide the Content-Length HTTP header.	411

TCP Error Codes   Wasabi also supports the following TCP error codes:

◆ Unexpected EOF

◆ Broken Pipe

◆ Connection reset by Peer

"Always Consistent" Feature

Rather than providing "always consistent" operations, AWS S3 operations are "eventually consistent." When you perform an AWS S3 operation, the results that follow may not reflect the previous operation.

Wasabi provides an "always consistent" view of all operations. Any operation following another will have the same expected result. This eliminates the special programming needed to wait for completion of a prior operation. As an example, after a DELETE on an object, it is common to perform a HEAD operation to check that the deletion was successful. Wasabi will always return a successful HEAD after the DELETE operation.

Authenticating Requests

Wasabi supports autonomous and signature versions 2 and 4 compatible with AWS S3, for both headers and query parameters. Generally, the signature version 4 offers better security, but requires more compute time on both the client and server because of the overhead calculating the SHA256 hash. If performance time is critical to your application, use the version 2 signing and check the calculated MD5 (returned as the ETag header for an object) for data integrity.

For version 4 signing requests, the region used for Wasabi signing requests is always "us-east-1" which is the default AWS S3 region.

Wasabi does not support signature signing for browser based uploads and HTML form requests because this is used for web page operations. (Web page operations are not currently supported on Wasabi).

For signature version 2, Wasabi added subresource keys that are specified as query parameters on the request. The specific subresource keys are:

◆ append

◆ compose

◆ force_delete=true

which are query parameters given for the PutObject and DeleteBucket operations. When using signature version 2, you need to include these in the calculation of the signing.

Not Supported in Wasabi

Operation	Description
SOAP	AWS S3 has deprecated support for SOAP. As a result, Wasabi does not support any SOAP operations.

3

Operations on Buckets

This section includes information for:

- DELETE BUCKET force_delete=true Option, page 3-2

- Renaming a Bucket, page 3-2

- MFA (Multi-Factor Authentication) Delete, page 3-2

- Maximum Number of Buckets, page 3-2

- Bucket Logging, page 3-3

- Bucket Cross-Origin Resource Sharing (CORS) Support, page 3-3

- Operations on Buckets Not Supported in Wasabi, page 3-4

- Compliance, page 3-4

DELETE BUCKET force_delete=true Option

AWS S3 will not allow you to delete a bucket if it contains objects that have not been deleted.

Wasabi provides a force_delete=true option that first deletes all the objects in the bucket and then deletes the bucket. The deletion of objects is subject to policy and compliance requirements on the bucket.

To specify the force_delete=true option, simply add it as a query string. For example:

```
DELETE http://s3.wasabisys.com/my-bucket?force_delete=true HTTP/1.1
```

Renaming a Bucket

AWS S3 does not support renaming of buckets. It only supports renaming of objects in a bucket.

Wasabi supports the renaming of buckets. The new bucket name must not be in use for the rename to be successful. The caller must have the s3:CreateBucket policy permission to rename a bucket.

To rename a bucket, use the HTTP method MOVE along with the header field "Destination" to give the new bucket name. For example:

```
MOVE http://s3.wasabisys.com/my_old_bucket HTTP/1.1
Destination: my_new_bucket
```

MFA (Multi-Factor Authentication) Delete

Wasabi supports the "x-amz-mfa" header while:

◆ configuring versioning on a bucket, or

◆ deleting objects with delete object requests compatible with AWS S3.

Wasabi does not require the "x-amz-mfa" header if the user's access credentials signing the request were authenticated with MFA. Wasabi only supports virtual MFA devices.

Maximum Number of Buckets

Standard AWS S3 supports only 100 buckets.

Wasabi allows for a maximum of 1000 buckets per account and this number may be increased by contacting Wasabi Customer Support.

Bucket Logging

Wasabi supports bucket logging, which creates a text log file of all access to a bucket. The format of the log file is identical to the AWS S3 log file.

Wasabi bucket logging does not require any ACL permission settings to store logs in a target bucket. Although you can give permission settings in the logging request or in an ACL, they are not required for logging to work in Wasabi. However, the bucket that is a target for log files must be inside the same account as the bucket being logged.

Bucket Cross-Origin Resource Sharing (CORS) Support

For compatibility with browser access to Wasabi as a web server, the Wasabi server will return CORS headers when the header "Origin" is given in an HTTP request. Additionally, the server supports the HTTP method OPTIONS on either buckets or objects to return the CORS headers needed for a browser pre-flight test before accessing Wasabi.

Different from AWS, Wasabi returns the settings that will allow the browser full access to Wasabi. Hence, Wasabi does not support the AWS functions that allow a PUT and GET on a bucket with the "cors" parameter in the URL. Note that allowing browser full access to data does not affect the security of access to any objects and all access policies will still be enforced.

The following are the HTTP headers returned by default when the header "Origin" is given in an HTTP request:

```
Access-Control-Allow-Headers: *

Access-Control-Allow-Methods: GET, HEAD, POST, PUT, DELETE, MOVE, OPTIONS

Access-Control-Allow-Origin: *

Access-Control-Expose-Headers: *

Access-Control-Max-Age: 86400
```

Operation	Description
Bucket LifeCycle	The ability to automatically delete objects after a certain time is provided in the Wasabi compliance feature described in Compliance, page 3-4. This compliance approach is viewed as superior to the bucket lifecycle policy used in AWS S3 (which is not currently available in Wasabi).
Bucket Replication	A bucket replication policy is currently unavailable in Wasabi. This feature is planned for a future release.
Bucket Tagging	Bucket tagging is currently unavailable in Wasabi.
Bucket Website	Website configuration is unavailable in Wasabi. Given the nature of Wasabi as a long-term object store, we do not expect to support website operations to buckets. The header "x-amz-website-redirect-location" is ignored in any object requests.
Bucket Accelerate	Wasabi does not implement the AWS S3 bucket accelerate subresource.
Bucket Location	Wasabi always returns the bucket location as "us-east". When creating a bucket, the "LocationConstraint" request data is ignored.
Bucket Notification	Wasabi does not support the use of the notification resource for a bucket.
Bucket Request Payment	Wasabi does not support the use of the "requestPayment" subresource for buckets.

Compliance   Wasabi supports a compliance policy that prevents the deletion of objects and provides additional informa-tion to prove that the original data is not modified since the time written. The compliance feature may be required for certain regulatory needs, but is also useful to prevent accidental data deletion.

You can set the compliance policy on any bucket controlling all the objects that are stored in that bucket. Specify the bucket compliance policy with the following XML tags.

Tag	Description
Status	Either "enabled" or "disabled" to turn compliance on and off, respectively. Enabling will immediately apply to all objects in the bucket.
LockTime	The time at which the compliance settings are "locked"—the settings cannot be reduced by any API call. Once the settings are locked, they cannot be unlocked without the intervention of Wasabi Customer Support. The lock time allows you to support two use cases: 1) testing that your software works properly before locking the compliance feature; or 2) never locking which means that data can be deleted with an additional step of an administrator turning compliance off. The lock time parameter may be: ◆ an ISO date (for example, 2016-11-07T15:08:05Z), ◆ the string "now" to force immediate locking, or ◆ the string "off to not lock the compliance settings. This is the default.
RetentionDays	An integer for the minimum number of days that objects are always retained after their creation date or release from conditional hold. You can extend the retention date for any individual object, but may not shorten the date. This parameter is always required.
ConditionalHold	A Boolean value ("true" or "false") indicating if newly created objects are placed on conditional hold, meaning that they cannot be deleted until the conditional hold is explicitly turned off. The default is false if this parameter is not given. Note that this setting may be changed even after the settings are locked.
DeleteAfterRetention	A Boolean value indicating if the object should be deleted automatically at the end of the retention period. The default is to not delete objects after the retention period. Note that this setting may be changed even after the settings are locked.

The compliance settings for a bucket are specified using the "?compliance" query string along with the compliance settings as the XML body in the request. For example:

```
PUT http://s3.wasabisys.com/my-bucket?complianceHTTP./1.1

<BucketComplianceConfiguration>

    <Status>enabled</Status>

    <LockTime>off</LockTime>

    <RetentionDays>365</RetentionDays>

    <DeleteAfterRetention>true</DeleteAfterRetention>

</BucketComplianceConfiguration>
```

After compliance is enabled for a bucket, the policy is immediately applied to all objects in the bucket. An attempt to delete an object before the retention period will return an error.

The compliance settings for a bucket can be retrieved by getting the bucket with the "?compliance" query string. For example:

```
GET http://s3.wasabisys.com/my-buck?complianceHTTP/1.1
```

Response body:

```
<BucketComplianceConfiguration xml ns="http://s3.amazonaws.com/doc/2006-03-01/">

    <Status>enabled</Status>

    <LockTime>2016-11-07T15:08:05Z</LockTime>

    <IsLocked>false</IsLocked>

    <RetentionDays>0</RetentionDays>

    <ConditionalHold>false</ConditionalHold>

    <DeleteAfterRetention>false</DeleteAfterRetention>

</BucketComplianceConfiguration>
```

There are also compliance settings for each object described in Operations on Objects.

4 Operations on Objects

This section includes information for:

Renaming Objects

Wasabi supports functionality to move an object-in effect, renaming the object by changing the key. This eliminates the two-step process of first copying an object and then deleting the original object. The caller must have s3:PutObject policy permission on the bucket to rename objects.

To rename objects, use the HTTP method MOVE along with the following parameters in the request headers that affect the move operation.

Parameter	Description
Destination	Used to form the destination key value for moved objects. If the X-Wasabi-Prefix option is true, then all objects (including all versions) that start with the source key in the URL are renamed by first removing the source key prefix and replacing it with this destination value. Otherwise, the objects (including all versions) exactly matching the source key in the URL are renamed to the key given by the destination. This parameter must be provided.
Overwrite	A Boolean value that, when "true," allows overwriting destination objects with the same key. Otherwise, an error is generated and the original key is not changed. Note that versioned buckets will always allow creating a new object version with the destination key. The default value is "false."
X-Wasabi-Quiet	A Boolean value that, when "true," causes the XML status return body to only display the keys that encounter errors. Otherwise, all objects renamed are given in the status. The default value is "false."
X-Wasabi-Prefix	A Boolean value that, when "true," means that the values given for the source in the URL and the destination are prefixes—they match the left-most part of keys. Conceptually, think of the prefix as a folder of objects. Otherwise, only the objects (including all versions) that exactly match the key are renamed. The default value is "false."

The call will return the results as an XML body of the response. The results include the following for each object renamed:

◆ original source key,

◆ renamed destination key,

◆ version ID of the object, and

◆ any error in the rename operation.

The quiet option will cause the results to list only the keys that encountered an error.

Below is an example, which renames all objects that start with the prefix "TestMove-Dir2/" to have the new prefix "TestMove-Dir2-Renamed/".

```
MOVE http://s3.wasabisys.com/my_bucket/TestMove-Dir2/HTTP/1.1

Destination:TestMove-Dir2-Renamed/X-BlueArchive-Prefix:true
```

Response body:

```
<MoveObjectResult>

    <SourcePrefix>TestMove-Dir2/</SourcePrefix>

    <DestinationPrefix>TestMove-Dir2-Renamed/</DestinationPrefix>

    <MoveObject>

        <SourceKey>TestMove-Dir2/</SourceKey>

        <DestinationKey>TestMove-Dir2-Renamed/</DestinationKey>

    </MoveObject>

    <MoveObject>

        <SourceKey>TestMove-Dir2/Dir1/</SourceKey>

        <DestinationKey>TestMove-Dir2-Renamed/Dir1/</DestinationKey>

    </MoveObject>

    <MoveObject>

        <SourceKey>TestMove-Dir2/Dir1/Obj1</SourceKey>

        <DestinationKey>TestMove-Dir2-Renamed/Dir1/Obj1</DestinationKey>

    </MoveObject>

    <MoveObject>

        <SourceKey>TestMove-Dir2/Dir1/Obj2</SourceKey>

        <DestinationKey>TestMove-Dir2-Renamed/Dir1/Obj2</DestinationKey>

    </MoveObject>

    <MoveObject>

        <SourceKey>TestMove-Dir2/Dir2/</SourceKey>
```

```
                    <DestinationKey>TestMove-Dir2-Renamed/Dir2/</DestinationKey>

                </MoveObject>

                <MoveObject>

                    <SourceKey>TestMove-Dir2/Dir2/Obj1</SourceKey>

                    <DestinationKey>TestMove-Dir2-Renamed/Dir2/Obj1</DestinationKey>

                </MoveObject>

                <MoveObject>

                    <SourceKey>TestMove-Dir2/Dir2/Obj2</SourceKey>

                    <DestinationKey>TestMove-Dir2-Renamed/Dir2/Obj2</DestinationKey>

                </MoveObject>

                <MoveCount>7</MoveCount>

            </MoveObjectResult>
```

Composing Objects

Wasabi provides a feature to create a new object that is composed of other objects. A composed object is formed not by copying the original data objects, but rather by linking the composed object to the data objects. This is in contrast to AWS S3, which does not offer a way to compose objects. Wasabi operation is much faster and does not require multiple copies of the original data. The original composed objects may be deleted at any time, and the system will retain the original data as long as there are any links to the data.

Composing objects is an alternative to using the multi-part upload feature for creating objects larger than 5 GB. The caller must have s3:PutObject policy permission on the bucket to compose objects.

Composed objects may be created from other composed objects as well as original data objects. However, the total number of original data objects may not exceed 1024 in any one composed object. Composed objects may only link to other objects in the same bucket.

The compose operation is done using the HTTP method PUT with the query string parameter "?compose" to indicate that the object is composed of objects given in the XML body. The XML body contains a list of object keys and, optionally, the version IDs that form the new object. If the version ID is not given, the latest version of the object is used. The new object will appear to be one object with all the linked data objects concatenated.

The new composed object does not have an ETag (i.e, MD5) value, which is the MD5 of the concatenated objects, but rather the result of hashing all the data object MD5s together (like the calculation done when multi-part objects are completed).

Composed objects may not have user provided encryption keys. However, all data is stored encrypted at rest.

There is a charge only for the metadata used in composed objects. The original data used in linked objects is charged at the normal rate until all composed objects that link to the original data object are deleted.

Below is an example that composes a new object "TestCompose-Object-1" from three objects: "TestCompose-Data-1", "TestCompose-Data-2", and "TestCompose-Data-3".

```
PUT http://s3.wasabisys.com/my_bucket/TestCompose-Object-1?composeHTTP/1.1Content-
Type:text/xml

<ComposeRequest>

    <Component>

    <Key>TestCompose-Data-1</Key>

    </Component>

    <Component>

    <Key>TestCompose-Data-2</Key>

    </Component>

    <Component>

    <Key>TestCompose-Data-3</Key>

    </Component>

</ComposeRequest>
```

Appending to Objects

AWS S3 does not offer a way to append to objects.

Appending to an existing object is a different form of composing an object (see Composing Objects). The call uploads the data in the body to a temporary object and then creates a new composed object consisting of the data from the original object with the newly uploaded data appended. If the bucket is versioned, a new version of the composed object is created while keeping the original data. When not versioned, the newly composed object replaces the original object. The caller must have s3:PutObject policy permission on the bucket

to append to objects. Additionally for a bucket without versioning, the s3:DeleteObject policy permission may be required to replace the original object.

Like composed objects, you can append only 1023 times to any object since each creates a link to the uploaded data. Be careful not to simply append small amounts of data many times to an object since each append data uploaded creates a new object subject to minimum size charges, and can slow down the performance due to a small data read.

The append operation is done using the HTTP method PUT with the query string parameter "?append". The data to be appended is uploaded in the body of the request.

Below is an example that appends the string "<appended data>" to an existing object "TestAppend-Object".

```
PUT http://s3.wasabisys.com/my-bucket/TestAppend-Object?appendHTTP/1.1Content-Length:15

<appended data>
```

PUT Object Copy Uses Link

Normally, to rename objects using AWS S3 protocol, an object is uploaded with one key value. Then, a PUT operation with the new key value copied from the original key value is given, and the object for the original key value is deleted.

Wasabi automatically optimizes this sequence by linking the new key value object to the original data from the copy key value. This avoids making another copy of the data, which would result in higher storage charges. Because Wasabi uses links to the original data, any size object up to the 5 TB limit may be copied and does not require the use of a multi-part upload.

Delete Object MFADelete

If the MFADelete option is turned on in the bucket versioning, the "x- amz-mfa" is not required if the access credentials were signed using MFA. (See also MFA (Multi-Factor Authentication) Delete, page 3-2.)

Additionally, the compliance feature can automatically delete objects when the DeleteAfterRetention option is true.

If bucket logging is enabled, deleting multiple objects will create a separate log entry for each object that is deleted.

GET Object Response Headers

Any header in the GET object request that starts with "response-" will be returned as a response header (minus the string "response-").

Object Server-Side Encryption

Wasabi encrypts all data stored at rest regardless of the requested encryption. The system will use any caller provided encryption keys when given, or generate a random key encryption key for each object if no customer key is provided. If the customer provides the encryption key, similar to AWS S3, Wasabi will not keep a copy of the customer key in the metadata and the caller must provide the encryption key to read the data. No action on the part of the caller is needed if Wasabi provides the encryption key.

The caller may provide the encryption key using the headers x-amz-server-side-encryption- customer-algorithm, x-amz-server-side-encryption-customer-key, and x-amz-server-side- encryption-customer-key-MD5. These parameters work identical to AWS S3.

Wasabi does not support a key management service. Hence, the "x-amz-server-side-encryption" is not supported along with all the "aws:kms" functionality.

Wasabi stores an MD5 for the data that is always the uploaded data regardless of server-side encryption.

Object Storage Class

Wasabi only provides a single storage class which is most like the standard AWS S3 storage class. Where the storage class is returned in any operation, Wasabi will return the standard AWS S3 storage class.

Operations on Objects Not Supported in Wasabi

Operation	Description
Object BitTorrent	Wasabi does not support BitTorrent operations on objects.
Object Tagging	Wasabi does not have the recently announced object tagging feature implemented. This may be added in a future release.
Object RESTORE	Wasabi does not support multiple classes of storage and does not support the POST object RESTORE request used to restore objects from a different class of storage.

Compliance   The compliance settings for any one object in a bucket with compliance can also be changed within the limits of the compliance on the bucket. Specify the object compliance settings with the following XML tags.

Tag	Description
ConditionalHold	A Boolean value "false" to release the object from the conditional hold setting in the bucket policy. The retention period in days is started from the point when the conditional hold is released. Once the conditional hold is set false, it may not be returned to conditional hold.
LegalHold	A Boolean value "true" or "false" to set the legal hold status. When an object has a legal hold status of true, the object cannot be deleted regardless of the retention period.
RetentionTime	An ISO time giving a new retention time for the object in which the object cannot be deleted before this time. Note that the new retention time must be past the retention period given by the bucket policy or an error is returned.

The following is an example of setting the compliance on an object:

```
PUT http://s3.wasabisys.com/my-bucket/my-object?complianceHTTP/1.1

<ObjectComplianceConfiguration>

    <ConditionalHold>false</ConditionalHold>

    <RetentionTime>2018-03-13T10:45:00Z</RetentionTime>

</ObjectComplianceConfiguration>
```

The compliance settings for any specific object can also be retrieved using the "?compliance" query string. In addition to the object compliance settings above, the query returns the calculated SHA256 hash for the object, which can be used to determine that the object has not been modified. Note that the SHA256 value is only available for objects that are uploaded as a single object and is not available for multi-part or composed objects.

The following is an example of getting the compliance on an object:

```
GET http://s3.wasabisys.com/my-bucket/my-object?complianceHTTP/1.1
```

Response body:

```xml
<ObjectComplianceConfiguration xml ns="http://s3.amazonaws.com/doc/2006-03-01/">
    <RetentionTime>2016-10-31T15:08:05Z</RetentionTime>
    <ConditionalHold>false</ConditionalHold>
    <LegalHold>false</LegalHold>
    <SHA256>14b4be3894e92166b508007b6c2e4fb6e88d3d0ad652c76475089a50ebe6e33b</SHA256>
</ObjectComplianceConfiguration>
```

The object compliance settings also appear in the bucket listings when the bucket has compliance enabled.

5 IAM and STS Support

Wasabi supports the following subset of the AWS S3 IAM:

AddUserToGroup	DeleteUser	ListEntitiesForPolicy
AttachGroupPolicy	DeleteUserPolicy	ListGroups
AttachRolePolicy	DeletetVirtualFADevice	ListGroupPolicies
AttachUserPolicy	DetachGroupPolicy	ListGroupsForUser
ChangePassword	DetachRolePolicy	ListMFADevices
CreateAccessKey	DetachUserPolicy	ListPolicies
CreateAccountAlias	EnableMFADevice	ListPolicyVersions
CreateGroup	GetAccessKeyLastUsed	ListRolePolicies
CreateLoginProfile	GetAccountAuthorizationDetails	ListRoles
CreatePolicy	GetAccountPasswordPolicy	ListUsers

CreatePolicyVersion	GetAccountSummary	ListUserPolicies
CreateRole	GetGroup	ListVirtualMFADevices
CreateUser	GetGroupPolicy	PutGroupPolicy
CreateVirtualMFADevice	GetLoginProfile	PutUserPolicy
DeactivateMFADevice	GetPolicy	PutRolePolicy
DeleteAccessKey	GetPolicyVersion	RemoveUserFromGroup
DeleteAccountAlias	GetRole	ResyncMFADevice
DeleteAccountPasswordPolicy	GetRolePolicy	SetDefaultPolicyVersion
DeleteGroup	GetUser	UpdateAccessKey
DeleteGroupPolicy	GetUserPolicy	UpdateAccountPasswordPolicy
DeleteLoginProfile	ListAccessKey	UpateAssumeRolePolicy
DeletePolicy	ListAccountAliases	UploadLoginProfile
DeletePolicyVersion	ListAttachedGroupPolicies	UpdateGroup
DeleteRole	ListAttachedRolePolicies	UpdateUser
DeleteRolePolicy	ListAttachedUserPolicies	

Wasabi supports the following subset of the AWS S3 STS.

◆ AssumeRole

◆ GetCallerIdentity

◆ GetSessionToken

All other actions in AWS S3 not listed are not support by Wasabi. All actions listed above are supported fully compatible with AWS S3 unless otherwise noted.

Wasabi supports only IAM actions for version "2010-05-08". The version value may be omitted from the call as well but, if given, it must be "2010-05-08". Wasabi supports only STS actions for version "2011-06-15". The version value may be omitted from the call as well but, if given, it must be "2011-06-15".

FORM Requests

Wasabi supports the use of the HTTP method POST with the content type "application/x-www- form-urlencoded" for IAM operations. The key/value data in the POST body is equivalent to using the data as query string parameters. For IAM requests with over 8000 characters of data, the request must be provided as a POST using the form encoded data.

Policy Compatibility

Wasabi supports access policies that are compatible with AWS S3 in definition. However, policies usually cannot be copied directly from AWS S3 to Wasabi because the policies frequently contain identifiers specific to the system (such as the account identifier and canonical IDs). The policy can be copied and all identifiers updated to their specific Wasabi values.

Policy Variables

Wasabi supports the following policy variables used in access policies:

aws:CurrentTime	aws:accountid (Wasabi feature only; see details below*)	s3:x-amz-content-sha256
aws:EpochTime		s3:x-amz-copy-source
aws: MultiFactorAuthPresent	iam:PolicyArn	s3:x-amz-grant-read
aws: MultiFactorAuthAge	s3:authtype	s3:x-amz-grant-write
aws:principaltype	s3:delimiter	s3:x-amz-grant-read-acp
aws:Referer	s3:max-keys	s3:x-amz-grant-write-acp
aws:SecureTransport	s3:prefix	s3:x-amz-grant-full-control
aws:SourceIp	s3:signatureAge	s3:x-amz-metadata-directive
aws:UserAgent	s3:signatureversion	s3:x-amz-server-side-encryption

| aws:userid | s3:VersionId | s3:x-amz-storage-class |
| aws:username | s3:x-amz-acl | sts:ExternalId |

* AWS S3 does not support a variable to return the account identification for the current users. Wasabi supports the additional policy variable "aws:accountid", which returns the account identification for the current user.

Multi-Factor Authentication (MFA) Support

Wasabi strongly encourages the use of an MFA device for additional security on your account. Wasabi supports virtual MFA devices, but not hardware MFA devices. Wasabi has been tested with Google Authenticator and Authy 2-Factor Authentication. However, Wasabi should operate with any application that supports the open TOTP standard.

System-Wide Policies

Wasabi provides some additional commonly used system-wide policies. The following are the standard system-wide policies and their meaning.

Policy	Description
WasabiS3FullAccess	Same as that defined by AWS, allowing full access to any AWS S3 resource.
WasabiS3ReadOnlyAccess	Same as that defined by AWS, allowing read only access to any AWS S3 resource.
AdministratorAccess	Same as that defined by AWS S3, allowing full administrator access to the account.
WasabiReadOnlyAccess	Allows read only access to any account, user, and AWS S3 data in the account. This policy also allows the user to change the password and use MFA.
WasabiWriteOnlyAccess	Allows the user to write objects to an AWS S3 bucket.
WasabiFullAccess	Allows the user full access to AWS S3 buckets and to modify their own user IAM parameters.

Policy Wildcard Matching

Wasabi supports these wildcard characters:

* can match zero or more characters.

? matches any one character.

Wildcards can match in the principal, action, resource, or conditions parts of the policy statement. These characters may be used anywhere in a string, but may not cross the colon (:) separator in an ARN. Prefixing the asterisk (*) and question mark (?) with a backslash (\) overrides the wildcard meaning of the characters and forces a match to the literal character.

When matching an ARN for an IAM resource or principal containing a path, matching is allowed either with or without the path given. However, if the policy ARN name contains a path, it must match the entity path.

Temporary Access Keys

Similar to AWS S3, Wasabi supports both permanent and temporary access credentials. Wasabi provides the action "CreateTemporaryAccessCredentials" for creation of temporary access credentials. Unlike other IAM actions, this action is not signed by credentials since it is used to obtain temporary credentials based on the user password. However, it must be protected using a secure HTTPS connection.

The action takes the following input parameters:

Parameter	Description
Account	A string that identifies the account using the account ID (with or without hyphens), account name (usually the email), or account alias. This parameter must be provided.
UserName	A string that identifies the user within the account. If no value is given, the user is assumed to be the root user for the account.
Password	A string that provides the password for the given user.
MFAToken	A string that provides the MFA token from the MFA device to authenticate the request when the user is required to use multi-factor authentication. This value may be omitted if no MFA token is required for the user. The authentication credentials retain the MFA authorization and can be used to avoid MFA authorization in other operations such as deleting objects.

Parameter	Description
Expires	An integer that provides the number of seconds until the access key expires. The maximum allowed is 129600 (which is 36 hours). If no value is provided, the default expiration time is 12 hours.
SecurityToken	A string that provides the security token to be used with the credentials. AWS S3 allows an optional security token that functions like a password to be used with the access key. If not provided, no security token is associated with the access keys.

The action returns in an XML body the access key ID, create time, expiration time, and the secret key for the access credential.

Below is an example call to the action "CreateTemporaryAccessCredentials":

```
GET https://iam.wasabisys.com/

?Action=CreateTemporaryAccessCredentials&AccountId=100000000100

&UserName=test-user&Password=my-passwordHTTP/1.1
```

Response body:

```
<CreateTemporaryAccessCredentialsResponse xmlns="https://iam.amazonaws.com/doc/2010-05-08/">

    <CreateTemporaryAccessCredentialsResult>

    <AccessKey>

    <AccessKeyId>AITXP7RW3QXQPXJ0XC9A</AccessKeyId>

    <CreateDate>2017-06-07T16:39:11.274Z</CreateDate>

    <Expires>2017-06-08T04:39:11.273Z</Expires>

    <SecretAccessKey>pKSYVSDH2aac1LVE8m4a6L070NA0MzOnTqWHzD2F</SecretAccessKey>

    </AccessKey>

    <User>

    <Path>/</Path>

     <PasswordLastUsed>2017-06-07T16:38:48.000Z</PasswordLastUsed>

    <Arn>arn:aws:iam::100000000103:root</Arn>
```

```
        <UserId>ADC37066E05B318487B6AEF5542E9C78FACC77A67CAD6B2FDC3990C430D15591</UserId>

        <CreateDate>2017-06-07T16:38:48.000Z</CreateDate>

        </User>

        <LoginProfile>

        <UserName />

        <CreateDate>2017-06-07T16:38:48.000Z</CreateDate>

        <PasswordResetRequired>false</PasswordResetRequired>

        </LoginProfile>

        <AccountPlan>

        <BillingPlanName>trial-premium</BillingPlanName>

        <Standing>ok</Standing>

        </AccountPlan>

        </CreateTemporaryAccessCredentialsResult>

        <ResponseMetadata>

        <RequestId>08a3add0-7ebc-903c-a9c6-23dccfeef920</RequestId>

        </ResponseMetadata>

    </CreateTemporaryAccessCredentialsResponse>
```

Change Password for Root User

Wasabi allows the root user to change the password using the action "ChangePassword". The access credentials identify the user and may include the account root user.

Create User Allows Setting Password

Wasabi allows the user password to be set with the action "CreateUser". The caller can provide the additional parameters "Password" and "PasswordResetRequired", which are the same as given for the "SetLoginProfile" action.

Getting User MFA Status

Wasabi returns more data about MFA status for an access key with the new action "GetUserMFA". The response data includes the MFA device data for the access key including the MFA device data for the user and the authorization date and age (in seconds) of the credentials. Note that the action may be called for roles in which case only the authorization date and age are returned. Below is an example call and response for the "GetUserMFA" action:

```
GET https://iam.wasabisys.com/?Action=GetUserMFA
```

Response body:

```
<GetUserMFAResponse xml ns="https://iam.amazonaws.com/doc/2010-05-08/">
    <GetUserMFAResult>
    <UserName>test-user-1</UserName>
    <SerialNumber>arn:aws:iam::100000000100:mfa/test/mfa-device-1</SerialNumber>
    <CreateDate>2016-11-15T15:29:39.223Z</CreateDate>
    <EnableDate>2016-11-15T15:29:39.228Z</EnableDate>
    <AuthenticateDate>2016-121-15T15:29:39.232Z</AuthenticateDate>
    <AuthenticateAge>15</AuthenticateAge>
    </GetUserMFAResult>
    <ResponseMetadata>
    <RequestId>2f967bbd-0b00-17a5-1907-69c37526cbd6</RequestId>
    </ResponseMetadata>
</GetUserMFAResponse>
```

Feature: Listing All Access Keys for Account

Wasabi allows the action "ListAccessKeys" to list all the access keys for the given account. If the username parameter is an asterisk (*), access keys are given for all the users, including the root user, in the account. The returned value of the username for the root user is blank.

Validating Policies   Wasabi supports the action "ValidatePolicy" to debug and test IAM policies. The action will return error information about any problems in the policy, such as incorrect structure and unrecognized identifiers. If there are no errors in the policy, the system will evaluate the policy for either an ALLOW, DENY, or no return status. The evaluation is done assuming the principal (account, user, or role) that signed the request.

The action takes the following input parameters:

Parameter	Description
PolicyDocument	The text of the policy document to evaluate. The text should be formatted as JSON.
IsPrincipal	The Boolean value "true" if this policy is applied to a principle element (a user, role, or group). Otherwise, the policy is assumed to apply to a specific resource (most often a bucket). AWS S3 differentiates from these two types of policies.
UseAction	The string giving the action for evaluating the policy. Example actions include s3:PutObject and iam:ChangePassword.
UseResource	The string giving the resource for evaluating the policy. Note that the resource should be expressed as an ARN (for example, arn:aws:s3:::my-bucket/my-object).
UseService	The service associated with the request: iam, sts, or s3. The default is iam.

6 Compatible Libraries and Applications

Wasabi continually performs interoperability testing with a wide range of storage management tools, gateways, and related products. The interop testing results are provided at wasabi.com/help. Because Wasabi is 100% bit-compatible with Amazon S3, any tool, gateway, or related product that is S3-compatible should work with Wasabi. Please contact support@wasabi.com if:

◆ You have questions about how to configure a particular product to work with Wasabi.

◆ You have questions about the interop results from a product not listed in the results table at wasabi.com/help.

◆ You are a product vendor and would like to conduct interop testing with Wasabi.

About This Guide

The *Wasabi API Guide* describes variations in functionality as they pertain to using the Wasabi API relative to Amazon Web Services' Simple Storage Service (AWS S3) and Identity and Access Management (AWS IAM) APIs. Wasabi's object storage service is built to be 100% bit-compatible with the AWS S3 and IAM APIs. In certain cases, Wasabi provides additional functionality relative to AWS S3 (or implements functionality in a different manner than AWS S3).

Content

Information in this guide is subject to change without notice. Companies, names, and data used in examples herein are fictitious unless otherwise noted. No part of this guide may be reproduced or transmitted in any form by means electronic or mechanical, for any purpose, without express written permission of Wasabi Technologies, Inc.

Technical Support

The Wasabi web site has the latest product information. Use this resource to review product information, download software updates, contact technical support, access the Wasabi knowledge base, or provide feedback on Wasabi documentation and online help:

www.wasabi.com

Use this email address to contact a Wasabi Customer Support representative:

support@wasabi.com