

S3 Object Key and Metadata

Each Amazon S3 object has data, a key, and metadata. Object key (or key name) uniquely identifies the object in a bucket. Object metadata is a set of name-value pairs. You can set object metadata at the time you upload it. After you upload the object, you cannot modify object metadata. The only way to modify object metadata is to make a copy of the object and set the metadata.

Object Keys

When you create an object, you specify the key name, which uniquely identifies the object in the bucket. For example, in the Amazon S3 console (see [AWS Management Console](#)), when you highlight a bucket, a list of objects in your bucket appears. These names are the object keys. The name for a key is a sequence of Unicode characters whose UTF-8 encoding is at most 1024 bytes long.

1024

Note

If you anticipate that your workload against Amazon S3 will exceed 100 requests per second, follow the Amazon S3 key naming guidelines for best performance. For information, see [Request Rate and Performance Considerations](#).

Object Key Naming Guidelines

Although you can use any UTF-8 characters in an object key name, the following key naming best practices help ensure maximum compatibility with other applications. Each application may parse special characters differently. The following guidelines help you maximize compliance with DNS, web safe characters, XML parsers, and other APIs.

Safe Characters

The following character sets are generally safe for use in key names:

- Alphanumeric characters [0-9a-zA-Z]
- Special characters !, -, _ , ., *, ' , (, and)

The following are examples of valid object key names:

- 4my-organization
- my.great_photos-2014/jan/myvacation.jpg
- videos/2014/birthday/video1.wmv

Note that the Amazon S3 data model is a flat structure: you create a bucket, and the bucket stores objects. There is no hierarchy of subbuckets or subfolders; however, you can infer logical hierarchy using key name prefixes and delimiters as the Amazon S3 console does. The Amazon S3 console supports a concept of folders. Suppose that your bucket (admin-created) has four objects with the following object keys:

Development/Projects1.xls

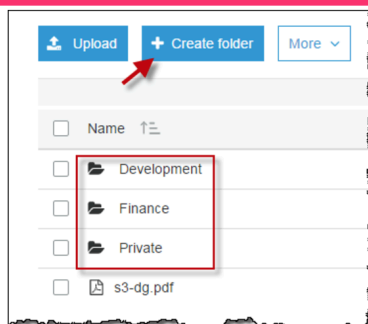
Finance/statement1.pdf

Private/taxdocument.pdf

s3-dg.pdf

Key name prefix and delimiter

The console uses the key name prefixes (Development/, Finance/, and Private/) and delimiter (/) to present a folder structure as shown:



The s3-dg.pdf key does not have a prefix, so its object appears directly at the root level of the bucket. If you open the Development/ folder, you see the Projects.xlsx object in it.

Object Metadata

There are two kinds of metadata: system metadata and user-defined metadata.

System-Defined Metadata

For each object stored in a bucket, Amazon S3 maintains a set of system metadata. Amazon S3 processes this system metadata as needed. For example, Amazon S3 maintains object creation date and size metadata and uses this information as part of object management.

There are two categories of system metadata:

- Metadata such as object creation date is system controlled where only Amazon S3 can modify the value.
- Other system metadata, such as the storage class configured for the object and whether the object has server-side encryption enabled, are examples of system metadata whose values you control. If your bucket is configured as a website, sometimes you might want to redirect a page request to another page or an external URL. In this case, a webpage is an object in your bucket. Amazon S3 stores the page redirect value as system metadata whose value you control. When you create objects, you can configure values of these system metadata items or update the values when you need to. For more information about storage classes, see [Storage Classes](#). For more information about server-side encryption, see [Protecting Data Using Encryption](#).

The following table provides a list of system-defined metadata and whether you can update it.

Name	Description	Can User Modify the Value?
Date	Current date and time.	No
Content-Length	Object size in bytes.	No
Last-Modified	Object creation date or the last modified date, whichever is the latest.	No
Content-MD5	The base64-encoded 128-bit MD5 digest of the object.	No
x-amz-server-side-encryption	Indicates whether server-side encryption is enabled for the object, and whether that encryption is from the AWS Key Management Service (SSE-KMS) or from AWS-Managed Encryption (SSE-S3). For more information, see Protecting Data Using Server-Side Encryption .	Yes
x-amz-version-id	Object version. When you enable versioning on a bucket, Amazon S3 assigns a version number to objects added to the bucket. For more information, see Using Versioning .	No
x-amz-delete-marker	In a bucket that has versioning enabled, this Boolean marker indicates whether the object is a delete marker.	No
x-amz-storage-class	Storage class used for storing the object. For more information, see Storage Classes .	Yes
x-amz-website-redirect-location	Redirects requests for the associated object to another object in the same bucket or an external URL. For more information, see (Optional) Configuring a Webpage Redirect .	Yes
x-amz-server-side-encryption-aws-kms-key-id	If the x-amz-server-side-encryption is present and has the value of <code>aws:kms</code> , this indicates the ID of the AWS Key Management Service (AWS KMS) master encryption key that was used for the object.	Yes
x-amz-server-side-encryption-customer-algorithm	Indicates whether server-side encryption with customer-provided encryption keys (SSE-C) is enabled. For more information, see Protecting Data Using Server-Side Encryption with Customer-Provided Encryption Keys (SSE-C) .	Yes

User-Defined Metadata

When uploading an object, you can also assign metadata to the object. You provide this optional information as a name-value (key-value) pair when you send a PUT or POST request to create the object. When you upload objects using the REST API, the optional user-defined metadata names must begin with "x-amz-meta-" to distinguish them from other HTTP headers. When you retrieve the object using the REST API, this prefix is returned. When you upload objects using the SOAP API, the prefix is not required. When you retrieve the object using the SOAP API, the prefix is removed, regardless of which API you used to upload the object.

Note

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

When metadata is retrieved through the REST API, Amazon S3 combines headers that have the same name (ignoring case) into a comma-delimited list. If some metadata contains unprintable characters, it is not returned. Instead, the `x-amz-missing-meta` header is returned with a value of the number of the unprintable metadata entries.

User-defined metadata is a set of key-value pairs. Amazon S3 stores user-defined metadata keys in lowercase. Each key-value pair must conform to US-ASCII when you are using REST and to UTF-8 when you are using SOAP or browser-based uploads via POST.

Note

The PUT request header is limited to 8 KB in size. Within the PUT request header, the user-defined metadata is limited to 2 KB in size. The size of user-defined metadata is measured by taking the sum of the number of bytes in the UTF-8 encoding of each key and value.

8 KB
2 KB

Adding a user defined Metadata to an Object

6. Type a custom name following the `x-amz-meta-` key. For example, for the custom name `alt-name`, the metadata key would be `x-amz-meta-alt-name`. Enter a value for the custom key, and then choose **Save**.

The screenshot shows a 'Metadata' dialog box with a blue header and a close button. Below the header are links for '+ Add Metadata', 'Delete', 'Edit', and an information icon. A table with two columns, 'Key' and 'Value', is visible. A 'New Metadata' button is present. The 'Key' field contains 'x-amz-meta-alt-name' and the 'Value' field contains 'sample-pic'. At the bottom are 'Cancel' and 'Save' buttons, with a red arrow pointing to the 'Save' button.

The screenshot shows the Amazon S3 console interface. At the top, the bucket name '1505038342893' is displayed with a 'Latest version' dropdown. Below this are tabs for 'Overview', 'Properties', and 'Permissions'. The 'Properties' tab is active, showing three sections: 'Storage class' (Standard), 'Encryption' (None), and 'Tags' (0 Tags). A 'Metadata' dialog box is open, showing the 'New Metadata' form. The 'Key' field is set to 'x-amz-meta-some-key' and the 'Value' field is set to 'some value'. Below the form are radio buttons for 'Content-Type' (application/octet-stream) and 'Website-Redirect-Location' (http://mahtabalam.net). At the bottom are 'Cancel' and 'Save' buttons.

Adding a System Defined Metadata to an Object

The screenshot shows the Amazon S3 console interface. At the top, the bucket name '1505038342893' is displayed with a 'Latest version' dropdown. Below this are tabs for 'Overview', 'Properties', and 'Permissions'. The 'Properties' tab is active, showing three sections: 'Storage class' (Standard), 'Encryption' (None), and 'Tags' (0 Tags). A 'Metadata' dialog box is open, showing the 'New Metadata' form. The 'Key' field is set to 'Website-Redirect-Lo...' and the 'Value' field is set to 'http://mahtabalam.net'. Below the form are radio buttons for 'Content-Type' (application/octet-stream) and 'Website-Redirect-Location' (http://mahtabalam.net). At the bottom are 'Cancel' and 'Save' buttons.

