SEAGATE

Lyve Cloud Object Storage Product Features

LYVE™ Cloud

Click here to access an up-to-date online version
of this document. You will also find the most recent content as well as expandable illustrations, easier navigation, and search capability.

Contents

1 Overview 6

Storage management 6
Access management and security 7
- Access management 7
- Security 7
- Availability and durability 7

2 Basic Information 9

Endpoint URLs 9
- About S3 REST API endpoints 9
- Lyve Cloud sites 9

Account IDs 10
Sign in to Lyve Cloud Object Storage 10

3 Product Features 12

S3 API support 12
Lyve Cloud console 12
- Console sessions 12
- Supported browsers 12

Multifactor authentication for GUI 13
- Set up MFA 13
- Disable MFA 14

Federated Login 14
Users and policies for S3 access control 15
- Use AWS defined policies 15
- Get an IAM policy file from AWS 15
- Use a policy permission file 16
- Example of policy permission file 18
- Create a policy with the JSON file 19

User accounts 20
Access keys 21
IP access security 22
Data at Rest Encryption (DARE) 22
Multiple region/sites 22
Inter-site geographical data replication 23
Presigned URL 23
Object tags 25
- Support for Zero-Byte Objects and Symlink Tags 27

Object versioning 27
WORM controls 28
Bucket retention policies 28
Object locking 29
- Retention periods 30
- Retention mode 30

- Legal holds . . . 31
- Best practices for object locking . . . 32
- S3 APIs supported . . . 32

Lifecycle configuration . . . 32
- Configure lifecycle rules . . . 32
- Remove lifecycle rules . . . 33
- Disable lifecycle configuration . . . 33

Lifecycle logic support . . . 33
- Lyve Cloud console support . . . 34
- S3 API support . . . 34
  - Lifecycle policy formatting . . . 35
  - Lifecycle for trimming previous versions of objects . . . 35
  - Lifecycle for removing unwanted objects . . . 36
  - Lifecycle for multipart cleanup . . . 38
  - Lifecycle for moving objects to Infrequent Access . . . 39
  - Moving objects back to STANDARD storage class . . . 40
  - Adding Lifecycles at the command line . . . 40

S3 audit/bucket logs . . . 42

IP source control . . . 47

CORS support . . . 48

Storage Class support . . . 49

4 Connect S3 Clients . . . 50

5 Connect Using AWS CLI . . . 51

6 Connect Using Cyberduck . . . 53

Prerequisites . . . 53

Connect Cyberduck to Lyve Cloud Object Storage . . . 53

Manage data . . . 55

Upload data to a bucket . . . 55

Download data to local storage . . . 56

Delete data from a bucket . . . 57

Create a new folder . . . 57

Delete a folder . . . 57

Disconnect Cyberduck from Lyve Cloud Object Storage . . . 57

7 Connect Using S3 Browser . . . 59

Prerequisites . . . 59

Connect S3 Browser to Lyve Cloud Object Storage . . . 59
- Advanced S3 compatible storage settings . . . 60

Manage data . . . 61

Upload data to a bucket . . . 62

Download data to local storage . . . 62

Delete data from a bucket . . . 63

Create a new folder . . . 63

Delete a folder . . . 64

Disconnect S3 Browser from Lyve Cloud Object Storage . . . 65

8 Connect Using Mountain Duck . . . 67

Prerequisites . . . 67
Connect Mountain Duck to Lyve Cloud Object Storage . . . 67
Manage data . . . 70
Copy data to a bucket . . . 70
Delete data from a bucket . . . 70
Create a folder in a bucket . . . 71
Disconnect Mountain Duck from Lyve Cloud Object Storage . . . 71

9 Connect Using Rclone . . . 73

Prerequisites . . . 73
Connect to Lyve Cloud from Linux . . . 73
- Install Rclone . . . 73
- Configure Rclone to connect to Lyve Cloud Object Storage . . . 73
- Mounting Lyve Cloud Object Storage as a drive . . . 78

Manage data . . . 78
- View information about your buckets and directories . . . 78
- Video: Lyve Cloud - How to use Rclone list commands . . . 79

Upload data to a bucket . . . 79
- Video: Lyve Cloud - How to use Rclone Copy-to and Copy-sync Commands . . . 79

Download data to local storage . . . 80
Delete data from a bucket . . . 80
Create a new folder . . . 80
Delete a folder . . . 81
Copy data . . . 81
- Deleting a remote from Lyve Cloud . . . 82

Migrate data . . . 82
- Video: Lyve Cloud - Use Rclone Delete and Purge commands . . . 82

10 .Frequently Asked Questions . . . 84

General . . . 84
Buckets . . . 88
Service Account . . . 92
User and Roles . . . 93
Self Service . . . 93
Billing . . . 95
Security . . . 97
Support . . . 100
Partner . . . 102
HIPAA . . . 103

11 .Lyve Cloud Compliance . . . 105

HIPAA . . . 105

- Lyve Cloud has a HIPAA Compliant report 105
ISO 27001 106
Type 2 SOC 2 106
- Lyve Cloud has a Type 2 SOC 2 Attestation report 106
Summary 107

Overview

Lyve Cloud Object Storage offers several features designed to support a variety of use cases. Customers can easily store, analyze, and manage data on secure, cost-efficient Seagate storage. Lyve Cloud provides an object storage solution that allows customers to move data to and from storage buckets through an HTTPS protocol. Admins can easily manage bucket access with user-specific access control lists. With a flexible application programming interface (API), customers can plug in their favorite S3-compatible applications to store data, run big data analytics, audit storage activity, and manage users across the platform.

Storage management

Lyve Cloud Object Storage breaks away from traditional storage classes to provide uninterrupted data movement. Objects can be uploaded, downloaded, updated, and erased anytime. Using S3 Select API calls, customers can easily connect to third-party clients to move and manage data. Applications are authenticated to Lyve Cloud Object Storage using an access key and secret key provisioned at service account creation. Once authenticated, applications will access buckets and objects using the defined permissions set in the service account (read-only, write-only, or all operations).

Bucket logs	All S3 API activity and actions are tracked with bucket logs. Bucket logs record all S3-supported API calls and activities on the console.
Object immutability and versioning	Lyve Cloud Object Storage offers features to help prevent unintended data modifications and provide versioning. Using object immutability prevents objects from being deleted or overwritten by any user, including the account owner, for a specified retention duration. Object immutability also supports Amazon S3 Object Lock to reinforce Write-Once-Read-Many (WORM) policies. Customers can enable object Immutability at bucket creation, which also activates versioning. Versioning allows customers to protect, recover, and restore every iteration of an object stored in a bucket in case of accidental deletions or failures. Versioning remains enabled even if object Immutability is later disabled.
Global account management	Global accounts allow customers to create buckets in different regions or service accounts to access different regional buckets. Lyve Cloud Object Storage also offers true replication of buckets across regions. This allows customers to have their data stored and accessible in multiple regions simultaneously. These functions provide simplified management of multiple regions on the console and the ability to increase redundancy and availability.

Lifecycle policies	The retention of data can also be controlled simply by the use of lifecycle policies. These enable customers to have the platform automatically manage the length of time that data objects are stored in the system.
Multi-level accounts	Users can create a multi-level account structure by using subaccounts to create, provision, and manage additional subaccounts. Each subaccount can be used to control who can access storage, create buckets, and upload data.

Access management and security

Access management

Account administrators have several tools to authorize access to Lyve Cloud Object Storage users. Identity and access management (IAM) allows administrators to manage users and their access to the console. Access is managed with user-defined roles that offer varying levels of accessibility. IAM users can use multi-factor authentication (MFA) for additional verification during login.

Configuring Federated Login requires Security Assertion Markup Language (SAML) protocol to provide a single sign-on authentication method through an organization's IDP (identity provider).

Security

Lyve Cloud Object Storage offers security features to protect data in flight and at rest. To ensure data is protected in flight, Lyve Cloud aligns with Transport Layer Security (TLS) 1.2 protocol and leverages 256-bit Advanced Encryption Standard (AES) Galois/Counter Mode (GCM) encryption, establishing secure communications to the client. By default, all data is encrypted before it is stored.

To learn more, see the Lyve Cloud Data Security Whitepaper.

Availability and durability

Lyve Cloud Object Storage data centers are located in multiple geographic locations, including Northern California, Virginia, Europe, and Singapore. Each location has dedicated operations staff to ensure services are available with a monthly uptime of 99.9%.

Data durability refers to long-term data protection against bit rot or other forms of corruption over long periods. Due to industry-leading architecture, Lyve Cloud Object Storage can achieve 11 9s of data durability making data loss statistically insignificant.

Basic Information

Endpoint URLs

You will be given URLs to access your service: one for GUI access, and another for S3 REST API access. There are other URLs which may be used for different linked services, but the GUI and REST URLs are the keys ones for most access. The URLs are referred to as your endpoint in this guide. They will typically take the following form:

GUI access	https://console.<endpoint-name>.lyve.seagate.com
REST access	https://s3.<site>.<endpoint-name>.lyve.seagate.com

Where <endpoint-name> appears in the URL, substitute the endpoint name value for your specific endpoint. Take note of your URLs, as they are your means of addressing the services over the internet.

Alternatively, you may receive a different URL as the endpoint for your service. This is a custom domain. In this case, you should use the custom URL in place of the <endpoint-name>.lyve.seagate.com wherever it appears in this guide. If needed, contact Lyve Cloud Support for more information.

About S3 REST API endpoints

Most applications and third-party systems will access the system via the S3 protocol. To do this, you will need the S3 REST API URL. This is always https://s3.<site>.<endpoint-name>.lyve.seagate.com, as mentioned above.

Lyve Cloud sites

The sites listed in the table below are available for customers of Lyve Cloud Object Storage. Site names can be used in the S3 requests and URLs specified above.

If your account doesn't currently have access to a site you want to use:

- Seagate direct customers—Use the Lyve Management Portal to manage available sites by modifying deployments in your Lyve Cloud subscription. The Lyve Management Portal is available at lyve.seagate.com.
- Service customers—If you purchased a service through a reseller, contact your account manager for help in accessing a site.

Region	Country or subregion	Site name
Asia and Pacific	Singapore	AP-SOUTHEAST-1
Asia and Pacific	Japan (Tokyo)	AP-NORTHEAST-1
Europe	Germany	EU-CENTRAL-1
Europe	United Kingdom	EU-WEST-1
United States	East coast	US-EAST-1
United States	Central	US-CENTRAL-1
United States	Central	US-CENTRAL-2
United States	West coast	US-WEST-1

Account IDs

An account ID is a unique identification that is associated with your Lyve Cloud Object Storage account. An account ID is unique across all Lyve Cloud accounts on your endpoint, and can include the company name specified during account creation. The account ID helps to identify and distinguish the resources in one account from the resources in another account.

> **i** The length of the account ID must be between 3 and 63 characters. Only lowercase characters, numbers, and hyphens `-` are allowed.

You cannot change the account ID once it is created.

The account ID is used in the login page as field in the information requested along with your username and password. The console URL has the following format:

https://console.<endpoint-name>.lyve.seagate.com/signin?=<account-ID>

Sign in to Lyve Cloud Object Storage

To use the Lyve Cloud Object Storage console, you must sign in using your account credentials. You will need a login URL, which contains a unique account ID. The login URL has the following format:

https://console.<endpoint-name>.lyve.seagate.com/signin?=<account-ID>

where <endpoint-name> is the unique name of your endpoint and <account-ID> is your unique Account ID.

Bookmark or save this URL to sign in to the console in the future.

Note that if you try to log in to the console using the simple URL https://console.<endpoint name>, you will be prompted to enter your unique Account ID.

After successful onboarding, you will receive a welcome email. This email contains the Lyve Cloud URL or endpoint for your access. Using the URL, you can sign in to Lyve Cloud Object Storage by creating a password. Select the **Forgot password** link to create the password.

Product Features

Lyve Cloud Object Storage provides a full set of features that allows customers to fully utilize the S3 Cloud Object Store in a variety of ways and with many third party applications. The sections below detail various features and functions of the system.

S3 API support

Lyve Cloud Object Storage offers a full range of API access by replicating AWS S3 REST commands and functions as much as possible, allowing you to use applications which have been validated against AWS without issue. See Lyve Cloud Object Storage API User Guide for details of supported REST API commands.

> **Note**—There are no limits on the number of buckets, objects, users, policies, or keys that an account can have in the system. However, the same limits for naming conventions and character sets are carried over from the S3 protocol.

Lyve Cloud console

Most system administrative functions can be performed using the Lyve Cloud console graphical user interface (GUI). The console allows admin and root users to:

- Manage users, policies, and access keys.
- Interact with buckets and manage bucket features and settings.
- View usage statistics for the account.

The features and functions exposed in the GUI are continually being expanded. For basic operational details, see the Lyve Cloud Object Storage Customer Guide.

Console sessions

The console remains active after successful authentication by the user and while interacting with the console. The user is automatically signed out after after a period of inactivity (default timeout is 15 minutes). The timeout setting can be set on a per user basis by selecting the **MY ACCOUNT** tab, which is accessed using the dropdown menu in the top right corner of the console.

Supported browsers

The console supports the following browsers:

Browser	Version
Google Chrome	Last three versions
Mozilla Firefox	Last three versions
Microsoft Edge	Last three versions
Apple Safari	Last three versions

Multifactor authentication for GUI

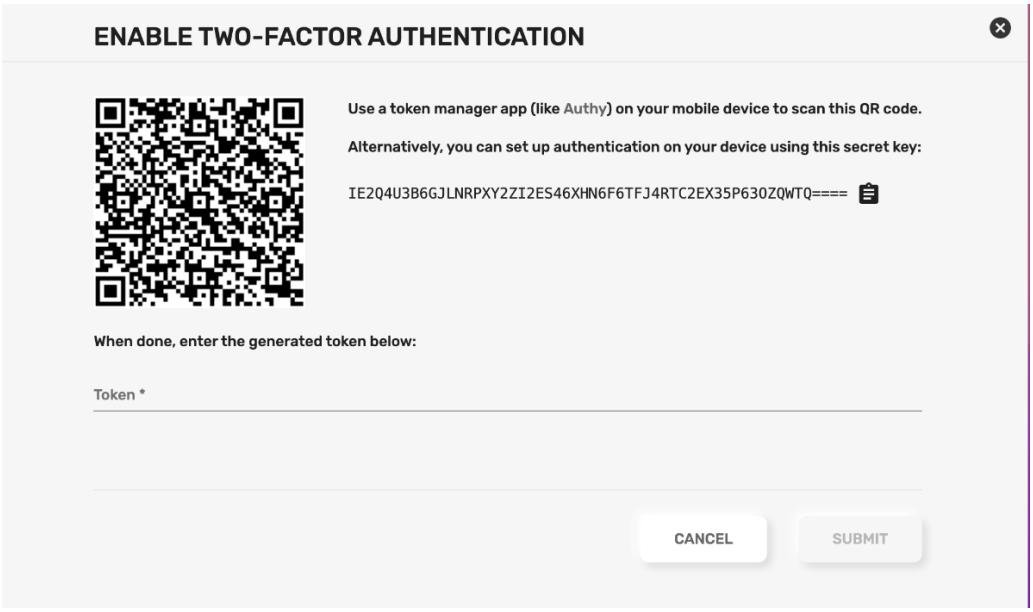
Users can set multifactor authentication (MFA) for their login. This is done on a per user basis through the Lyve Cloud console GUI on the**MY ACCOUNT** tab. The process supports most third-party authenticator applications.

Set up MFA

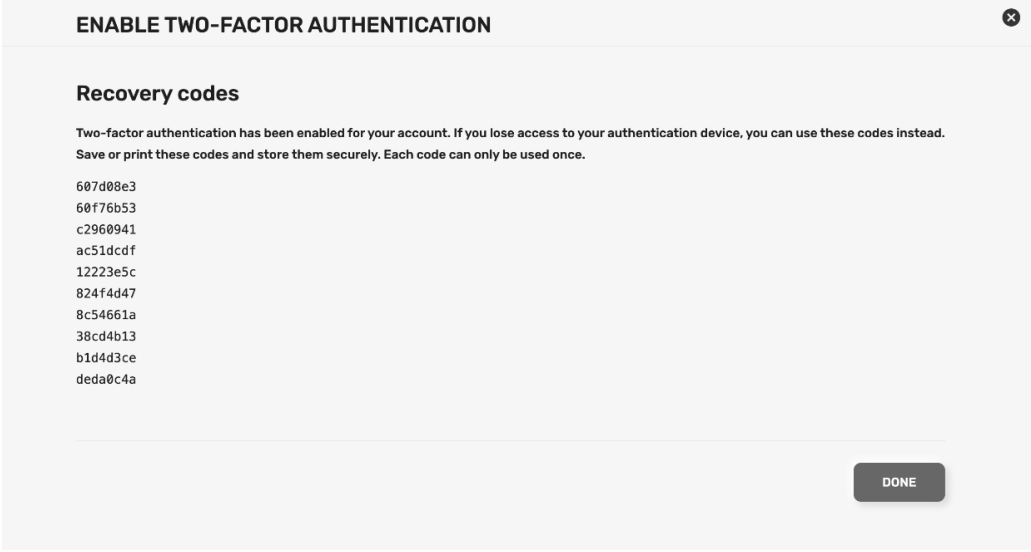
If you set up MFA to access the system every time you log in, you must follow the two-step authentication process. The console GUI requests users to enter a one-time password (OTP) generated from an authenticator app. An OTP is a unique password that is only valid for a single login session or transaction.

To set up MFA:

1. Log in to the console using your credentials.



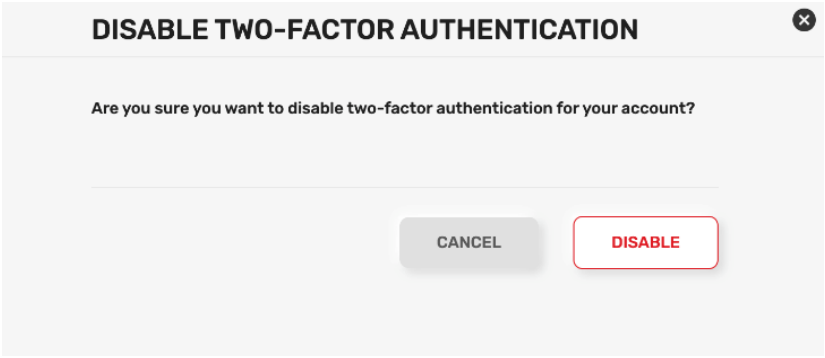
2. Use a third-party authenticator app such as Google, Microsoft, or Oracle Mobile Authenticator to generate an OTP. The authenticator app generates a random OTP that expires within a time limit.
3. In the console, enter the OTP displayed in the authenticator app.
4. Select **SUBMIT**.
5. After MFA has been enabed, Lyve Cloud Object Storage will generate a set of recovery codes. Save or print the recovery codes and store them in a secure location. In the event that you lose your authenticating device, you can use the recovery code to temporarily log in again.
6. Select **DONE**.



> **Note**—If you lose your recovery codes, contact Lyve Support at lyve.support@seagate.com.

Disable MFA

To disable MFA, select the **DISABLE** button in MY ACCOUNT.



Federated Login

You can configure a federated login system so that users only need to sign in to your organization's domain in order to have direct access to the Lyve Cloud console. To use the Federated Login feature, your

organization must have an authentication system which uses the Security Assertion Markup Language (SAML) 2.0 protocol. SeeConfiguring Federated Login for more details on how to enable this feature.

> **Note**—Federated Login is configured by a root or admin user via the Lyve Cloud console GUI. However, it is still possible to allow users normal password access users after Federated Login has been configured. In fact, it's recommended that the root user be left as a password access account.

Users and policies for S3 access control

Lyve Cloud Object Storage utilizes many of the standard AWS IAM commands for control of users, policies, buckets, and more, allowing easier integration with third-party storage management systems. For supported IAM REST commands, see theLyve Cloud Object Storage API User Guide

Permissions via policies are used to control who can access buckets and which actions they are allowed in a bucket. Bucket permissions are granted by assigning policies to user accounts.

Aside from the root user, each user needs a policy to access buckets and obejcts. Policies are assigned to users, and a user can have multiple policies. Each policy can be created/edited in the Lyve Cloud console using a simple interface for assigning**read**, **list**, **write**, and **delete** controls to a bucket. Alternatively, you can upload a JSON file with predefined policies. All policy administration can be managed using REST commands in addition to the console.

Use AWS defined policies

The system allows the migration of AWS IAM policies to Lyve Cloud Object Storage, making it simple to start working with service accounts based on existing policies. A policy file uses a JSON file format that is compatible with an AWS IAM policy.

Working with policy files allows you to:

- Specify the **Condition** element: Query the exact request values to determine when a policy is in effect, or list specific actions such asAction: ["s3:GetObject","s3:PuObject"].
- Specify the **Resource** element for several buckets and objects.

Get an IAM policy file from AWS

You can manually copy policy permission details from AWS IAM policy to use in the system. This is to replicate an existing AWS policy in the Lyve Object Storage system. To do this use the following steps:

1. Log in to AWS Management Console.
2. Select **Services** in the top left corner of the page to view the list of services.

3. Select **IAM** in 'Security, Identity, & Compliance'.
4. Under 'Access Management', select **Policies** and use the search field to find the relevant policy.
5. Select the **JSON** tab. Copy the policy details into a new file, and then save it as a JSON file.

> **Note**—Invalid elements must be removed from the file before importing, as these elements are not used in the Lyve Cloud policy permission file. Remove tags from elements available in AWS IAM policy, as tags cannot be used in the policy permission file.

Use a policy permission file

The following table lists elements in the policy permission file and specifies if which are mandatory, optional, or invalid.

Elements	Mandatory/Optional/Invalid	Description
Statement	Mandatory	Contains a single statement or an array of individual statements.
Resource	Mandatory	Specifies object(s) or bucket(s) that is related to the statement.
Effect	Mandatory	Allows or denies access to the resource.
Action	Mandatory	Describes specific action(s) that will be allowed or denied.
Version	Mandatory	It defines the version of the policy language and specifies the language syntax rules that are to be used to process a policy file.

Condition	Optional	Allows you to specify conditions when a policy is in effect. The **Condition** element includes expressions that match the condition keys and values in the policy file against keys and values in the request. Specifying invalid condition keys returns an error. For more information, see Known Issues.
Sid	Optional	A statement ID. The statement ID must be unique when assigned to statements in the statement array. This value is used as sub ID for policy document's ID.
Id	Optional	A policy identifier, such as UUID (GUID).
Principal	Invalid	Specifies the service account that is allowed or denied to access a resource.
NotPrincipal	Invalid	The service accounts that are not specified, are allowed or denied access to the resource.
NotAction	Invalid	Specifies that it matches everything except the specified list of actions. If this element is part of the permission file, you need to replace it with the Action element.

NotResource	Invalid	Specifies that it matches every resource except the available specified list. If this element is part of the permission file, you need to replace it with the resource element.

Example of policy permission file

In the following example, the policy permission has three statements:

- `Statement1`: Allows object listing with a prefix `David` in the bucket `mybucket`. It is done using a **Condition** element.
- `Statement2`: Allows read and write operations for objects with the prefix `David` in bucket `mybucket`.
- `Statement3`: Denies delete object operation for two resources:
  - All the objects in `mybucket/David/*`
  - All the objects in `mycorporatebucket/share/marketing/*`

```
{
   "Version": "2012-10-17",
   "Statement": [
      {
         "Sid": "statement1",
         "Action": ["s3:ListBucket"],
         "Effect": "Allow",
         "Resource": ["arn:aws:s3:::mybucket"],
         "Condition": {
            "StringLike": {
               "s3:prefix": ["David/*"]
            }
         }
      },
      {
         "Sid": "statement2",
         "Action": [
            "s3:GetObject",
            "s3:PutObject"
         ],
         "Effect": "Allow",
         "Resource": ["arn:aws:s3:::mybucket/David/*"]
      },
      {
         "Sid": "statement3",
         "Action": ["s3:DeleteObject"],
         "Effect": "Deny",
```

```
            "Resource": [
                "arn:aws:s3:::mybucket/David/*",
                "arn:aws:s3:::mycorporatebucket/share/marketing/*"
            ]
        }
    ]
}
```

The following policy limits bucket access to specific IP addresses:

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "Sid-1",
            "Action": ["s3:*"],
            "Effect": "Deny",
            "Resource": ["arn:aws:s3:::mybucket"],
            "Condition": {
                "NotIpAddress": {
                    "aws:SourceIp": ["134.204.220.36/32"]
                }
            }
        },
        {
            "Sid": "Sid-2",
            "Action": [
                "s3:*"
            ],
            "Effect": "Allow",
            "Resource": [
                "arn:aws:s3:::mybucket",
                "arn:aws:s3:::mybucket/*"
            ]
        }
    ]
}
```

Note—The IP Protect function can be used to protect the whole account rather than on a 'per bucket' basis.

Create a policy with the JSON file

To set a policy permission with the JSON file you just created:

1. In the top menu, select **Policies**.
2. On the Policies page, select the button on the left side of the page to create a new policy.

3. In the 'Create Policy Permission' dialog, enter a name.
4. Select **Upload JSON file** and then select the file.
5. Select **Submit**.

User accounts

User accounts associate keys with policies in the system. Keys obtain the policies and role permissions of the user account for which they're created. You must have at least one user account to create keys.

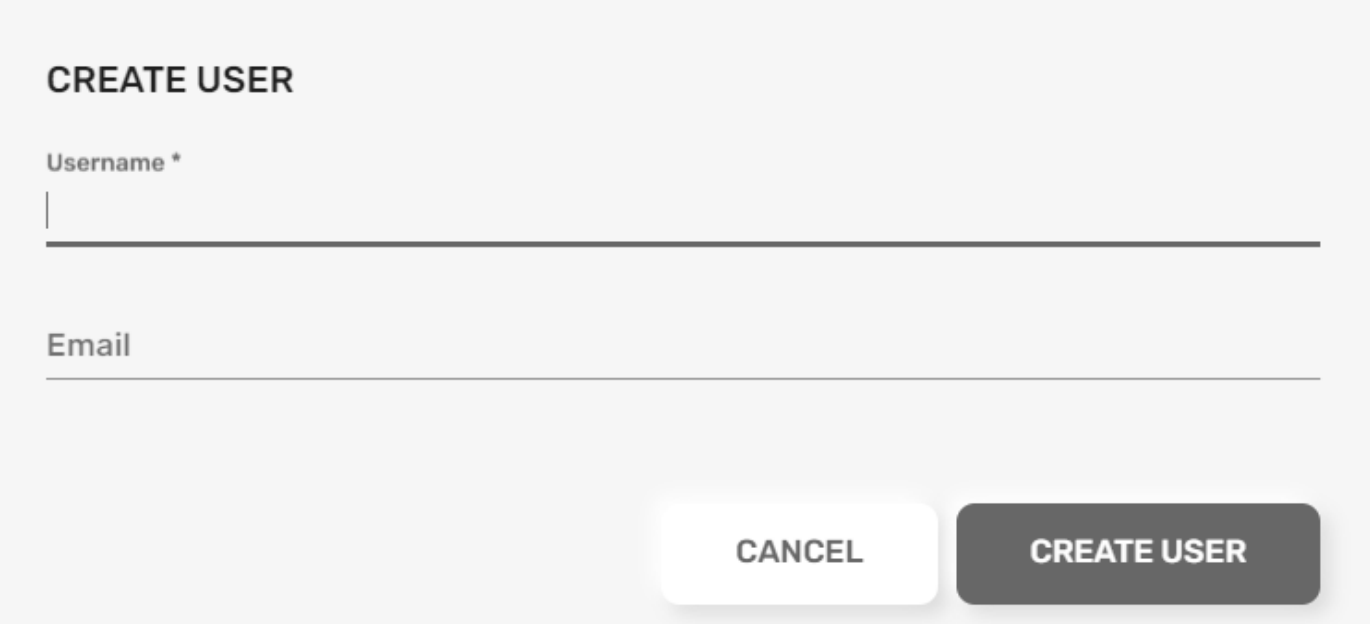
> **Note**—While the root user account can be used to create keys, this represents a security risk in that the generated keys will automatically have full access and permissions in the account.

Each account is assigned one of three roles in the system:

Role	Account level permissions	Bucket level permissions	Comments
Root	Can perform changes on account level settings as well as other users and policies	This user has access to all buckets	The account owner. This user can not be deleted or changed.
Admin	Can perform changes on account level settings as well as other users and policies. Can perform any REST operations supported in the account.	Policies must be attached for it to have access to buckets and objects.	The level of account is for administration functions in the account.
Sub-user	Not allowed to perform account level actions or make changes to users or policies. Limited account level REST operations allowed.	Policies must be attached for it to have access to buckets and objects.	Typically used for applications to access buckets and use the system.

To create a user:

1. Select the **User** tab on the top menu.
2. On the User page, select the button on the right side of the screen.
3. In the dialog, enter a username and an email address for the account.
4. Select **CREATE USER**.



Note the following items when creating/using users accounts:

- The **root** user is defined at account creation and cannot be changed.
- Each user can have multiple access keys and policies associated with it.
- All the values for username and the email must be entered in lower case.
- The username and the email address can be the same value. This is a simple method of creating users accounts who can also access the Lyve Cloud console.
- The email address is optional. However, if omitted, the account cannot be used to access the console. This type of user is typically created for application access where account access keys will be used for access to data.
- A welcome email message is sent to the user to set up their password when provisioned (if they have an email address).
- By default, accounts are created in a**sub-user** role. You can modify a created account to give it the **admin** role.
- The **root** role user can reset the password of any account in the system and generate keys for them. They can also modify any policy and assign those to any user.
- Any **admin** user can reset the password of any**admin** or **sub-user** account in the system and generate keys for them. They can also:
  - Modify any policy and assign it to any user.
  - Create lifecycle policies.
- A **sub-user** role user can only generate access keys and reset their own password. They can not modify policies or policy assignments.

Access keys

Access for third-party applications is provided with access keys. Keys do not expire but can be deleted using the Lyve Cloud console or S3 commands. This allows for manual rotation of keys as needed.

A key is associated to a specific user and uses the policies and role they are assigned for the permission settings.

In the console, access keys are shown in the user account, accessible via the Users page or at the top right under **MY ACCOUNT:**

- Use the button to generate new keys.
- View the access key value of existing keys.

> **Note**—Creating an access key automatically generates a secret key as well, which is used by the system in authentication processes. When creating an access key/secret key, the console provides you with an opportunity to download the pair in CSV format. This is a one-time opportunity—you can no longer view the secret key after you've exited the key creation sequence. If you lose the secret key information at a later time, the only option is to delete the key and create a new one for use.

IP access security

Lyve Cloud console and S3 REST command interactions use Transport Layer Security (TLS) for data in-flight: TLS 1.2+ (AES-256-GCM). This level of protection is also is used for all network traffic internal to the cloud object store.

See IP source control below.

Data at Rest Encryption (DARE)

Data at Rest Encryption (DARE) a process that encrypts data stored on physical media so that it can only be accessed by those with a key. SSE-S3 level of data storage and encryption is provided by Seagate, who manages the keys by default for all buckets. The system will accept explicit SSE-S3 encryption requests correctly in the REST API for storage, but will not change the default behavior of the system.

Lyve Cloud Object Storage also supports encryption with a client-provided key (provided as part of S3 request headers), commonly known as SSE-C. This uses keys provided by the client for data encryption. The system does not store or retain the keys used in these cases, so the client needs provide the same keys for data access or the data will not be accessible.

Multiple region/sites

Lyve Cloud Object Storage offers multiple regions/sites for data storage in the US, Europe, and Asia. Access to specific regions needs to be granted to your account to create buckets in them.

If you need access to locations which are not shown in your account, contact Lyve Support at lyve.support@seagate.com.

Inter-site geographical data replication

When creating a bucket, you can configure it to be automatically replicated on two or more regions/sites. Data is automatically saved and duplicated separately in each location. Each copy of the bucket is active and can be interacted with separately—there is no master/slave configuration, all copies of the bucket are live and active in this system.

The different regions/sites communicate constantly to ensure that a change in one location is almost instantly (usually within 1 second) known and available in all locations.

> **Note**—If concurrent access is required in a single bucket, it's strongly recommend that you use versioning (see below) to ensure that no data is lost.

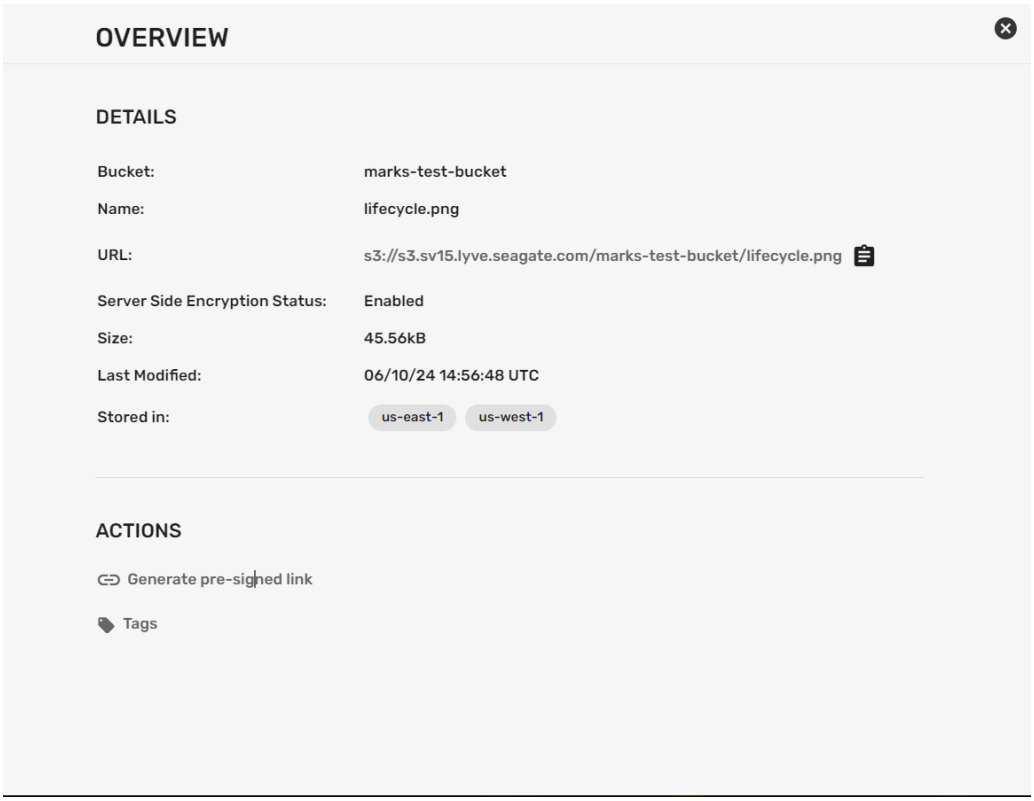
Inter-site data replication allows customers in different geographic location to access the same bucket while interacting with 'local' copies of the data. This removes the need for long distance connections to remote data sets and increases data availability—if one site is not accessible (for whatever reason), requests can be directed to another site to access the data.

There is no additional cost for this feature other than the cost for storage in each location.

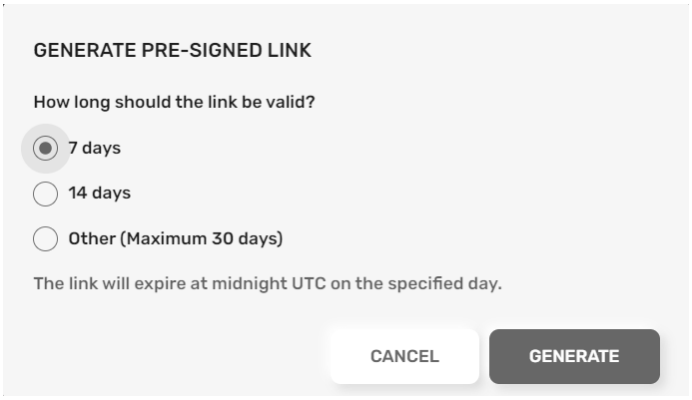
Presigned URL

Lyve Cloud Object Storage lets you create pre-signed URLs for objects for both GET and POST operations. The REST API fully supports all functionality for generating pre-signed links.

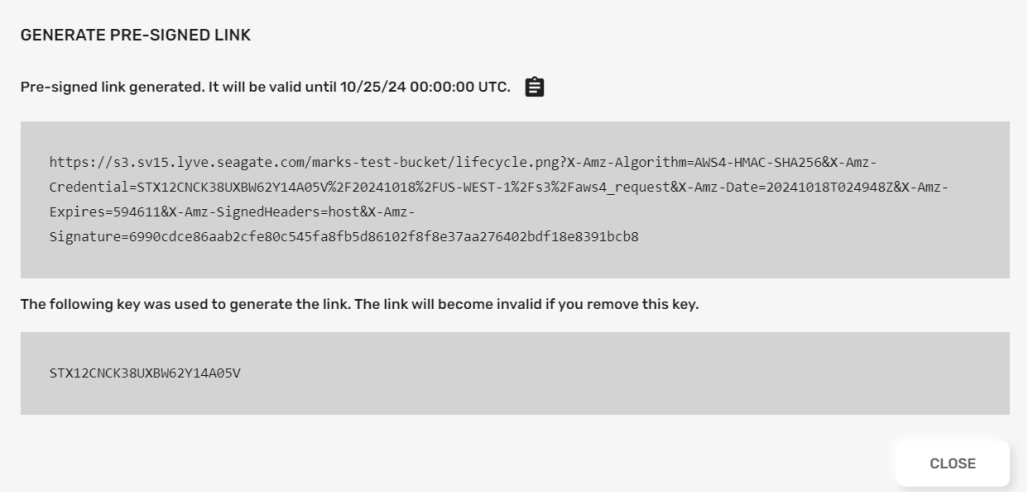
You can also create a pre-signed link by selecting a specific object in the Lyve Cloud console, and then selecting the **Generate pre-signed link** action.



In the dialog, specify how long the link will be valid for:, and then select **GENERATE**:



A summary displays detailed information on the generated link.



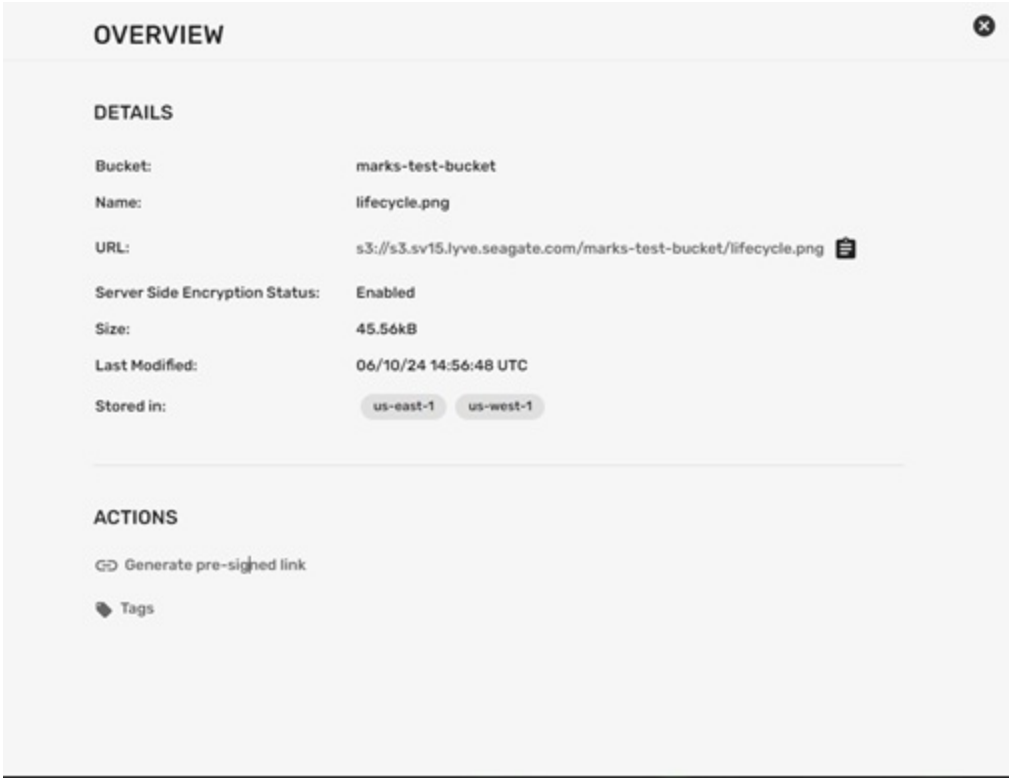
> **Note**—The link will be retained in the system for the specified period, but the detailed information is only available in the summary. Make sure to copy the details by selecting the Copy to Clipboard icon in the summary. If you lose the link details, another link will need to be generated.

Object tags

Tags let you store extra data items related to objects. The system supports up to 10 custom tags on each object. These are accessed and controlled via standard S3 API calls, but can also be viewed via the Lyve Cloud console.

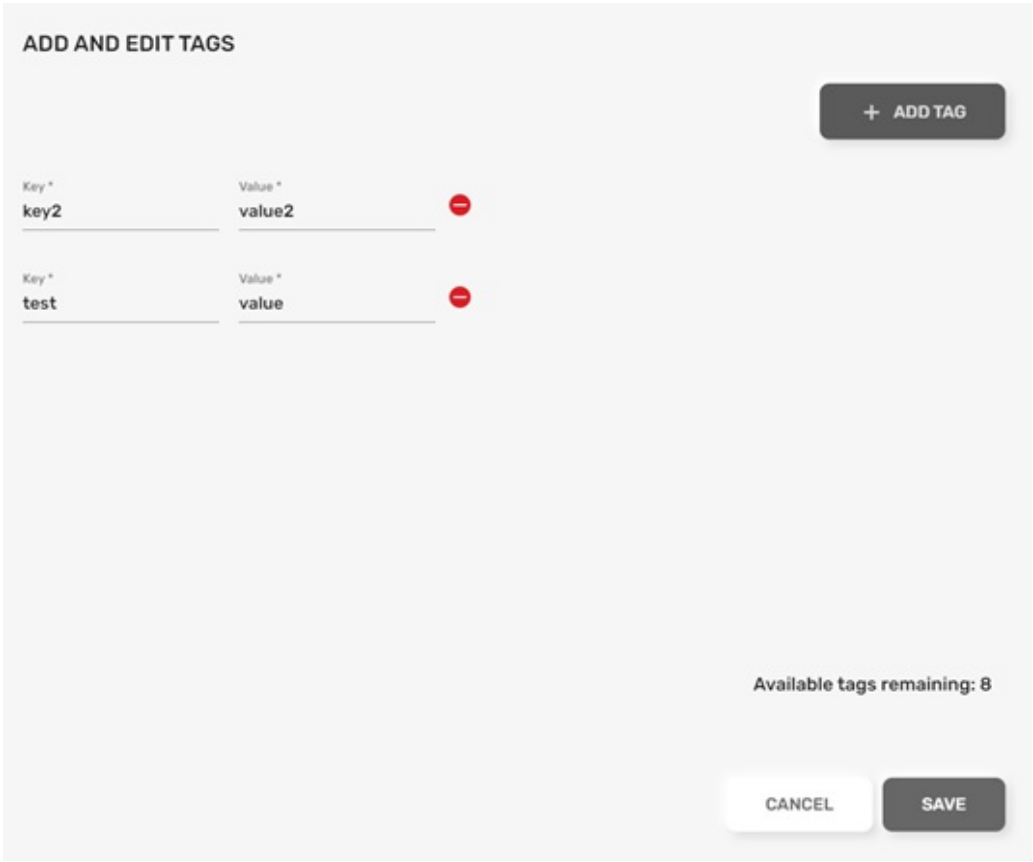
> **Note**—Tags are not indexed in the system and are not searchable via standard S3 REST commands or the GUI console

Tags can also be seen in the console by selecting the **SHOW DETAILS** button on an object, and then selecting the **Tags** link:



You can:

- View and edit key/value pairs for existing object tags.
- Select ADD TAG to add a new tag (limit 10 per object).
- Select the Delete icon to remove a tag.



Support for Zero-Byte Objects and Symlink Tags

Lyve Cloud supports the creation and storage of zero-byte objects, which can be useful for placeholder files or signaling mechanisms in workflows. Additionally, object tags can be used to simulate symbolic links (symlinks) by assigning metadata that references other objects. This enables more flexible data organization and access patterns, especially when integrating with third-party tools or custom applications.

Object versioning

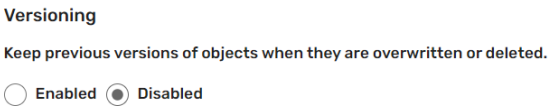
Object versioning provides protection from data loss. Versioning allows you to save multiple variants of an object in the same bucket. You can then preserve, retrieve, and restore any version of an object that was was in the bucket. Versioning enables the recovery of objects due to unintended user actions or accidental application failures.

In a bucket with versioning enabled, Lyve Cloud Object Store automatically creates and stores an object version whenever:

- A new object is uploaded
- An existing object is overwritten
- An object is deleted

For example, when you delete an object in a versioned bucket, the object isn't removed from the bucket—instead, 'deleted' is just the current version of the object, while the previous object is now just an older version of itself. In short, when versioning is enabled, all operations on existing objects are really just a history of changes.

Versioning has to be set when creating a bucket. In the Lyve Cloud console, enabling versioning is a simple radio button.



In a create bucket REST operation, versioning is a parameter.

> **Note**—Versioning cannot be modified for an existing bucket.

The downside of versioning is that buckets will grow in size as object histories grow. 'Deleting' an object will in fact increase the storage used, not reduce it. There are specific ways to delete the old versions of the object using the 'version_id' parameter in the request to identify the specific version of the object to

be removed. Applications which support versioned buckets (such as the Lyve Cloud console) will offer this option when handling versioned buckets.

Some applications, including the Lyve Cloud console, let you manage and control old versions of objects. You can also use the lifecycle logic to automatically control the length of time that version records will be retained in a versioned bucket.

WORM controls

When creating a bucket it can be set to be a Write-Once Read Many (WORM) bucket, often referred to as **object locking**. WORM means that data can be written to the bucket and accessed, but is not allowed to be deleted. If this option is set, then versioning is always automatically enabled for a bucket. WORM can be set using the Lyve Cloud console or via REST API calls.

WORM prevents objects from being deleted or overwritten in a bucket by any user or application. WORM can be set for a specified retention duration using the bucket retention policies detailed below. This functionality is especially useful when you want to meet regulatory data requirements, or other scenarios where it is imperative that data cannot be changed or deleted. This feature should be used when you are certain that you do not want anyone, including an administrator, to delete the objects during their retention duration.

Bucket retention policies can be added to the object-lock setting, but are not required. Some applications (especially backup systems) want to control that themselves, and so enabling this level is all that is required for them as they will set the retention rules on a per object level in the bucket.

> **Note**—Use of this feature means that you cannot use normal commands to delete objects created in buckets. This includes lifecycle policies and any other operations.

Implementation for WORM is done at the software level and must be specified when the bucket is created. It is not possible to change this setting after bucket creation.

Bucket retention policies

If WORM is enabled, you can configure the system to add restrictions for how long an object must be retained in the bucket. Retention policies can be set during bucket creation and to some level can be modified afterwards.

> **Note**—It is only possible to extend the limits of policies, not reduce them. These can be controlled via the Lyve Cloud console or REST API calls.

The duration for immutability (that is, the inability to delete) can be specified in days at the bucket level. You can set this value at the object level if required – which some applications do. When you set the duration, objects at the bucket level will remain locked and cannot be overwritten or deleted until that time period as passed. Setting the duration applies to individual object versions, and different versions of a single object can have different durations .

When you place an object in the bucket, the system calculates the retention duration for an object version by adding the specified duration to the object version's creation timestamp. The calculated date is stored in the object's metadata and protects the object version until the retention duration ends. When retention duration ends for an object, you can retain or manually delete an object.

The system supports both compliance and governance modes of data storage as defined by the AWS S3 system. In either case, the retention period is specified in days.

> **Note**—Use of this feature will mean that you cannot use normal commands to delete objects created in buckets with this setting.

Object locking

Object locking prevents objects from being deleted or overwritten by any user or application for a specified retention period. This is especially useful when you want to meet regulatory data requirements or when it is imperative that data cannot be changed or deleted.

To enable object locking:

1. Log in to the Lyve Cloud console.
2. Under Object Locking, select the **Enabled** radio button.

> Object locking only works in buckets that have object versioning enabled. When you enable object locking, versioning will automatically be enabled and cannot be disabled.

3. Under Default Retention, select one of the following:

- **Enabled**—A default retention period is automatically applied to all objects in the bucket.
- **Disabled**—Each object has a unique retention period.

4. If you enabled "Default Retention", specify the retention period and select a compliance mode. See Retention periods and Retention modes below.

Retention periods

A retention period is a fixed period during which an object version remains locked. You can specify a retention period between 1 day to 100 years. When a retention period is placed on an object version, the object version's metadata stores a timestamp to indicate when the retention period expires. After the retention period expires, the object version is essentially unlocked and can be deleted.

The retention period defines the number of days or years to protect the object version. When an object gets placed in the bucket, the 'Retain Until Date' for the object version is calculated by adding the retention period to the object version's creation timestamp.

Retention periods apply to individual object versions, and different versions of the same object can have different retention modes and periods. For example, if you set a retention period of 10 days and then create object A, it will have its retention period set to 10 days. If you later change the retention period to 20 days and upload the same object A again:

- The retention period for the first version of object A remains 10 days.
- The later version of the same object is set to 20 days.

Retention mode

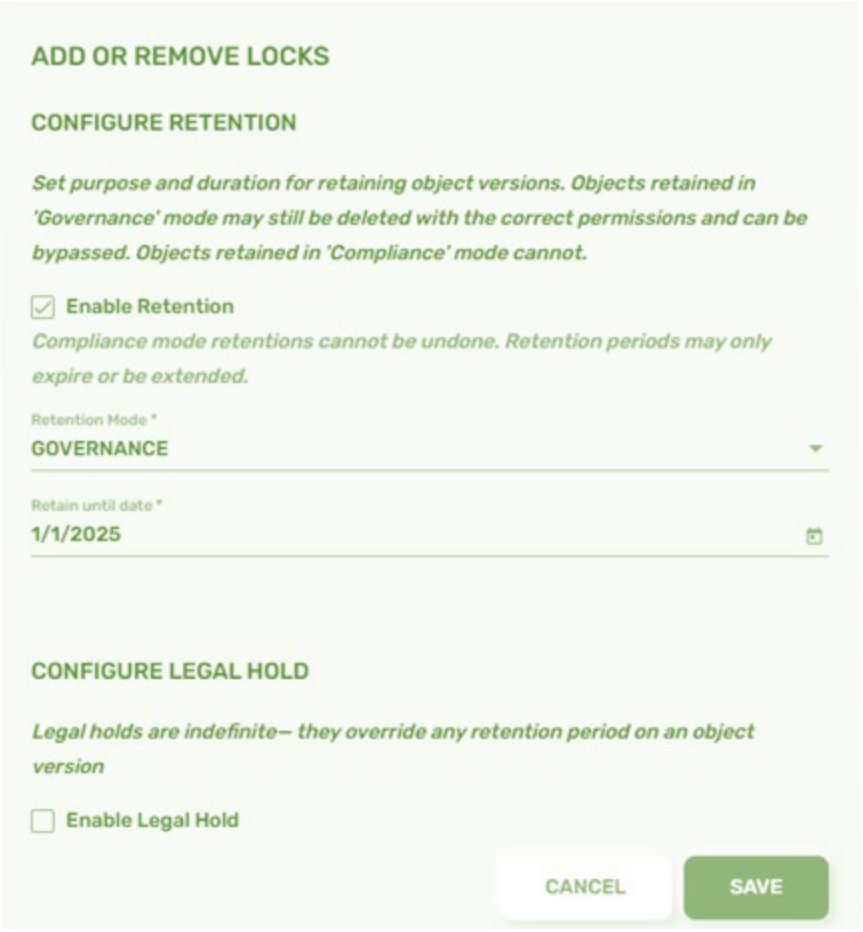
A retention mode is a setting to specify what levels of protection is required for your objects. Lyve Cloud supports two retention modes:

Compliance	Object versions cannot be overwritten or deleted by any user, including the root admin user of your Lyve Cloud account. When an object is locked in compliance mode, its retention mode cannot be changed and its retention period cannot be reduced.

Governance	Users cannot overwrite or delete an object version or alter its lock settings unless they have special permissions. With this mode, the objects are protected from being deleted by most users, but special permissions can be granted to some users so that they can change the retention settings or delete objects if necessary. The best use case for governance mode is to use it for testing retention-period settings before creating a compliance-mode retention period.

Legal holds

With object locking, you can also place a legal hold on an object version. In Lyve Cloud, you can enable legal hold by selecting the**Enable Legal Hold** checkbox on an object version. To remove the legal hold, deselect the checkbox.



Just like retention period, a legal hold prevents an object version from being overwritten or deleted. However, a legal hold does not have a fixed amount of time and will remain in effect until it's removed. Legal holds are indefinite by design—they override any retention period set on an object version. Legal holds can be placed and removed by users with special permissions.

Legal holds are independent from retention periods. Placing a legal hold on an object version does not

affect the retention mode or retention period for that object version. For example, if you place an object version on legal hold, the object version is also protected by retention period. Even if the retention period expires, the object version does not lose its protection, as the legal hold will continue to protect until the user explicitly removes the legal hold. Similarly, if the user removes a legal hold while an object version has retention period in effect, the object version remains protected until the retention period expires.

Best practices for object locking

- Consider using governance mode if you want to protect objects from being deleted by most users during a pre-defined retention period, but at the same time want some users with special permissions to have the flexibility to alter retention settings or delete objects.
- Consider using compliance mode if you don't want any user, including the root admin user of your Lyve Cloud account, to be able to delete objects during a pre-defined retention period. You can use this mode if you have a requirement to store compliant data.
- Use legal holds when you are not sure how long you want your objects to stay immutable. This could be because you have an upcoming external audit of your data and want to keep objects immutable until the audit is complete. Alternatively, you may have an ongoing project utilizing a dataset that you want to keep immutable until the project is finished.

S3 APIs supported

- PutBucketObjectLockConfiguration
- GetBucketObjectLockConfiguration
- PutObjectLegalHold
- PutObjectRetention
- GetObjectLegalHold
- GetObjectRetention

Lifecycle configuration

Lifecycle configuration allows you to create and manage lifecycle policies for objects stored in a bucket. A lifecycle policy consists of one or more lifecycle rules that define when objects transition to a different storage class or expire.

Configure lifecycle rules

Follow these steps to create and enable lifecycle rules for a bucket:

1. Log in to the Lyve Cloud console.
2. In the sidebar, select **Buckets**.
3. In the list of buckets, find the bucket you want to configure and click the wrench icon.
4. Select the **Lifecycle Configuration** tab.
5. Select the **Enable Lifecycle** checkbox.
6. In the **Lifecycle Rules** section, select **+ Add Lifecycle Rule**.
7. In the **Rule Details** section, enter a **Rule Name** to identify the lifecycle rule. The rule name must be

between 3 and 255 characters.

8. Under **Rule Scope**, choose one of the following options:
   - **Apply to all objects in the bucket** – The rule applies to every object stored in the bucket.
   - **Add filter to rule** – The rule applies only to objects whose names begin with a specified prefix.
9. If you selected **Add filter to rule**, enter a single object prefix in the field provided. The prefix is the beginning portion of an object name and is used to limit the rule to a subset of objects. Do not include the bucket name in the prefix.
10. Under **Rule Actions**, select one or more of the following actions:
    - **Transition current versions of objects to infrequent storage class** – Moves current object versions from the standard storage class to the infrequent access storage class after a specified number of days since object creation.
    - **Expire current versions of objects** – Permanently deletes current object versions after a specified number of days. Objects stored for less than 180 days in the infrequent access storage class may incur early deletion charges.
11. For each selected action, enter the number of days after object creation when the action should occur.
12. (Optional) Repeat these steps to add additional lifecycle rules.
13. Select **Save** to apply the lifecycle configuration to the bucket.

Remove lifecycle rules

1. Open the Lyve Cloud application.
2. In the sidebar, select **Buckets**.
3. In the list of buckets, find the bucket you want to configure and click the wrench icon.
4. Use the **Lifecycle Rules** dropdown to select the rule you want to delete.
5. In the **Rule Details** section, click the trash icon.
6. Select **Save** to apply the updated lifecycle configuration to the bucket.

Disable lifecycle configuration

1. Open the Lyve Cloud application.
2. In the sidebar, select **Buckets**.
3. In the list of buckets, find the bucket you want to configure and click the wrench icon.
4. Clear the **Enable Lifecycle** checkbox.
5. Select **Save** to disable lifecycle configuration for the bucket.

Lifecycle logic support

Lifecycle logic helps you manage object storage costs by automating actions throughout an object's lifecycle. You manage object lifecycles by creating **lifecycle policies** at the bucket level. Each bucket can have its own lifecycle policy—if configured, each bucket must have a policy.

You can manage lifecycle policies by using either the S3 API or the Lyve Cloud console. To modify bucket lifecycle policies, you must have admin-level or root user permissions.

An S3 Lifecycle configuration currently supports the following use cases:

Trim previous versions of objects	You set the limit for the age of old versions of objects to be removed. The system then automatically deletes object versions which are older than the specified time in an automated manner. This allows for easy control of versioned buckets without manual intervention and provides a safety net to prevent accidental overwrites or deletions without unlimited storage growth.
WORM/Lock object data deletion clean up	Allows automated cleanup of data after retention requirements expire in WORM buckets. Objects protected by retention rules cannot be deleted. By specifying deletion rules longer than the retention period, you can automatically remove objects after retention expires.
Archive data deletion	Automatically deletes objects that have exceeded a defined retention period, providing a simple way to clean up old or unused data.
Clean up unwanted data or buckets	Lifecycle policies can be used to delete large numbers of objects efficiently, which is especially useful when deleting buckets containing millions of objects.
Multipart cleanup	If you frequently perform large multipart PUT operations—especially over unreliable or slow network connections—uploads may be interrupted before completion. When this happens, partially uploaded object parts remain stored so the upload can be resumed or recovered. Although the system eventually cleans up these incomplete uploads, they continue to consume storage and increase costs until that cleanup occurs. You can use a lifecycle policy to delete incomplete multipart uploads earlier than the system's default cleanup window, helping reduce unnecessary storage usage and associated costs.
Move objects to Infrequent Access	Moves objects to Infrequent Access storage after a defined number of days, if your subscription supports the infrequent access storage tier.

Lyve Cloud console support

The Lyve Cloud console provides a simple interface for setting up the most common lifecycle policies. Admin users can use it to create simple rules for common use cases. See Lifecycle configuration above.

S3 API support

Lifecycle logic is supported through S3 REST API operations. The following API calls are used to manage lifecycle policies on a bucket:

- PutBucketLifecycleConfiguration
- GetBucketLifecycleConfiguration
- DeleteBucketLifecycle

These commands are detailed in theAPI reference documentation. Lifecycle logic follows the same concepts as Amazon S3 lifecycle policies. Below are some examples of how to set policies for different use cases.

Lifecycle policy formatting

A bucket can have only one lifecycle policy defined, but that policy can contain multiple rules. As a result, a single policy can be quite sophisticated and control multiple aspects of how objects in the bucket are managed over time.

The commands above apply a lifecycle policy, which is defined in JSON format. The examples below illustrate the available options and how they are used. One important consideration when defining rules is whether the bucket is versioned or non-versioned, as rule behavior can differ between the two. If this is a new concept, seeObject versioning for an explanation.

Lifecycle policies are typically authored in a JSON file and then referenced by the relevant commands. The policy must include specific required fields, along with one or more action definitions:

- Each policy rule must include an**ID**. This is a required, human-readable identifier used to distinguish the rule. The value can be any text string you choose.
- The **filter** field is also required. It defines which objects the rule applies to. In most cases, the filter is left empty, which causes the rule to apply to all objects in the bucket. The only other supported option is a prefix filter. When a prefix is specified, the rule applies only to objects whose names begin with that string. Prefix matching is case-sensitive.
- The **status** field controls whether the rule is active. A policy can contain multiple rules, and individual rules can be disabled by setting this field to**Disabled**.

Lifecycle for trimming previous versions of objects

For a versioned bucket, you can control how long old versions of overwritten or deleted objects are retained by using the following JSON policy. In the following example, replace**<days>** with the number of days you want previous object versions to be kept before they are permanently removed.

```
{
   "Rules": [
      {
         "ID": "Trim-non-concurrent-object-versions",
         "Filter": {},
         "Status": "Enabled",
```

```
            "NoncurrentVersionExpiration": {
                "NoncurrentDays": <days>
            }
        }
    ]
}
```

To keep old changed objects for 30 days, set **<days>** to be **30** as shown below:

```
{
    "Rules": [
        {
            "ID": "Trim-non-concurrent-object-versions",
            "Filter": {},
            "Status": "Enabled",
            "NoncurrentVersionExpiration": {
                "NoncurrentDays": 30
            }
        }
    ]
}
```

Lifecycle for removing unwanted objects

To remove all objects in a bucket, a lifecycle policy is the recommended approach. The logic differs slightly for versioned buckets, but for a non-versioned bucket—where you simply want to control how long objects are retained—use the following example. Replace **<days>** with the number of days you want objects to be kept before they are deleted.

> **i** Note that **1** is a valid value in Lyve Cloud, but **0** is not. If you specify **1**, the rule applies to all objects that are at least one day old.

```
{
    "Rules": [
        {
            "Expiration": {
                "Days": <days>
            },
            "ID": "Delete-after-X-days",
            "Filter": {},
            "Status": "Enabled"
        }
    ]
}
```

To delete objects after a year (365 days), set **<days>** to **365**:

```
{
   "Rules": [
      {
         "Expiration": {
            "Days": 365
         },
         "ID": "Delete-after-365-days",
         "Filter": {},
         "Status": "Enabled"
      }
   ]
}
```

For a versioned bucket, this rule does not immediately delete objects—it marks them as deleted. To fully remove objects from a versioned bucket, use this rule together with a rule that deletes non-current object versions.

To control both how long current objects are retained and how long old versions are kept, use the following example. In this case, **<days-current>** is the number of days to retain current objects, and **<days-previous>** is the number of days to retain previous versions. Replace these values as needed.

```
{
   "Rules": [
      {
         "Expiration": {
            "Days": <days-current>
         },
         "ID": "Delete-after-X-days-versioned",
         "Filter": {},
         "Status": "Enabled"
         "NoncurrentVersionExpiration": {
            "NoncurrentDays": <days-previous>
         }
      }
   ]
}
```

```
{
   "Rules": [
      {
         "Expiration": {
            "Days": 1
```

```
        },
        "ID": "Delete-after-1-day-versioned",
        "Filter": {},
        "Status": "Enabled"
        "NoncurrentVersionExpiration": {
            "NoncurrentDays": 1
        }
    }
  ]
}
```

As another example, to delete all objects in a versioned bucket after ten years—but retain a safety window in which deleted objects remain recoverable for 60 days—the policy would look like the following:

```
{
  "Rules": [
    {
      "Expiration": {
        "Days": 3650
      },
      "ID": "Delete-after-10-years-and-retain-version-60-days",
      "Filter": {},
      "Status": "Enabled"
      "NoncurrentVersionExpiration": {
        "NoncurrentDays": 60
      }
    }
  ]
}
```

Lifecycle for multipart cleanup

When a client uses multipart uploads, it may pause or delay completing the upload. In these cases, any parts already uploaded to Lyve Cloud remain stored until the upload either completes, the system removes them, or the client explicitly issues an abort request. Many applications do not send an abort, so the platform cleans up unfinished multipart uploads only after seven days.

If you have a large number of failed multipart uploads that were neither completed nor aborted, you can use a lifecycle policy to remove these partial uploads sooner than the default seven-day cleanup window. The example policy below removes partial uploads after two days:

```
{
  "Rules": [
    {
      "ID": "Delete-multipart-uploads",
      "Filter": {},
      "Status": "Enabled",
      "AbortincompleteMultipartUpload": {
```

```
            "DaysAfterInitiation": 2
        }
    }
  ]
}
```

Lifecycle for moving objects to Infrequent Access

Lyve Cloud supports multiple storage classes. If your billing plan includes the Infrequent Access storage tier, this is how those objects are identified. (If your plan does not include this option, contact Lyve Cloud Sales.) Be aware that Infrequent Access objects have specific retention and storage restrictions.

By default, all newly uploaded objects are stored in the STANDARD storage class. Lifecycle logic is the mechanism you use to transition objects to a different storage class.

There are several reasons to use lifecycle logic to transition objects to STANDARD_IA, the Infrequent Access storage class. Examples of each use case are provided below. Note that once an object has been moved to STANDARD_IA, the only way to return it to STANDARD is to copy the object over itself using the REST Copy operation and explicitly specify the STANDARD storage class. There is no lifecycle rule that promotes objects back from STANDARD_IA to STANDARD.

One common scenario is moving all objects in a bucket to the Infrequent Access tier. The rule below applies this transition to every object in the bucket:

```
{
  "Rules": [
    {
      "ID": "All objects should be infrequent access",
      "Filter": {},
      "Status": "Enabled",
      "Transition": {
        "Days": 1,
        "StorageClass": "STANDARD_IA"
      }
    }
  ]
}
```

If objects were moved to Infrequent Access after 90 days, use the following rule:

```
{
  "Rules": [
    {
      "ID": "All objects should be infrequent access after 90 days",
      "Filter": {},
      "Status": "Enabled",
```

```
            "Transition": {
                "Days": 90,
                "StorageClass": "STANDARD_IA"
            }
        }
    ]
}
```

Objects retain their storage-class setting after they are created, unless a lifecycle policy changes it. A less common use case involves customers who never want to delete objects but want all non-current versions to be stored in the Infrequent Access tier. The rule below accomplishes this: primary (current) object versions remain in the STANDARD storage class, while all non-current versions transition to Infrequent Access after one day.

```
{
    "Rules": [
        {
            "ID": "Noncurrent objects should be Infrequent Access after 1 day",
            "Filter": {},
            "Status": "Enabled",
            "NoncurrentVersionTransition": {
                "NoncurrentDays": 1,
                "StorageClass": "STANDARD_IA"
            }
        }
    ]
}
```

Moving objects back to STANDARD storage class

Once an object has been transitioned to STANDARD_IA, there is no automated method to return it to the STANDARD storage class. As soon as an object moves to STANDARD_IA, it begins billing under that tier and becomes subject to its associated minimum-retention requirements.

To move an object back to the STANDARD tier, you must issue a COPY request that creates a new object—using the same object name—and explicitly sets the storage class to STANDARD. This operation creates a new current version; it does not remove or shorten the remaining retention period of the previous version.

Adding Lifecycles at the command line

Using the AWS CLI, you can configure a lifecycle policy for the data in a bucket. Follow these streamlined steps:

1. **Install AWS CLI.** If it's not already installed, begin by installing AWS CLI. The CLI provides direct access to the S3 API, making it easy to manage S3-compatible resources. Installation instructions are available at https://docs.aws.amazon.com/cli/latest/userguide/getting-started-install
2. **Configure AWS CLI.** After installation, configure your credentials for S3 access. Replace **<your-**

access-key> and **<your-secret-key>** with your actual values. Leave the remaining fields blank if you are unsure.

```
aws configure
AWS Access Key ID []: <your-access-key>
AWS Secret Access Key []: <your-secret-key>
Default region name [None]:
Default output format [None]:
```

3. **Prepare the JSON policy file.** Create a JSON file containing the lifecycle policy you want to apply. The contents depend on the rules and actions you intend to use. Refer to the sections above for details on supported rule types.
4. **Apply the lifecycle policy.** Use the following command to apply the lifecycle policy to your bucket, replacing **<bucket-name>**, **<file>**, and **<URL-of-service>;** with your specific values:

```
aws s3api put-bucket-lifecycle-configuration --bucket <bucket-name> --lifecycle-configuration <file>.json --endpoint-url <URL-of-service>
```

5. **Allow time for processing.** Lifecycle policies generally begin taking effect within 48 hours. Full completion may take longer, depending on the number of objects in the bucket and how many changes the policy requires.

Implementing lifecycle policies is straightforward, but issues or questions may still arise. The points below cover common problems and important considerations:

AccessDenied errors: This message indicates the credentials you are using likely do not have the required permissions. Verify that your access keys have the correct administrative privileges.

Error Parsing Parameter: This message typically indicates an issue with the JSON policy file—most often a formatting error or an incomplete copy-and-paste. Review the JSON for syntax errors or missing characters.

Legal hold and retention policies: Objects that are under legal hold or subject to retention policies cannot be deleted by lifecycle rules. If expected deletions are not occurring, check the bucket's object-lock and retention settings.

Verifying lifecycle policies: To confirm that a lifecycle policy is applied to a bucket, run the command below, replacing **<bucket-name>** and **<URL-of-service>;** with your values. Note that the correct parameter for specifying the service URL is --endpoint-url.

```
aws s3api get-bucket-lifecycle-configuration --bucket <bucket-name> --endpoint-url <URL-of-service>
```

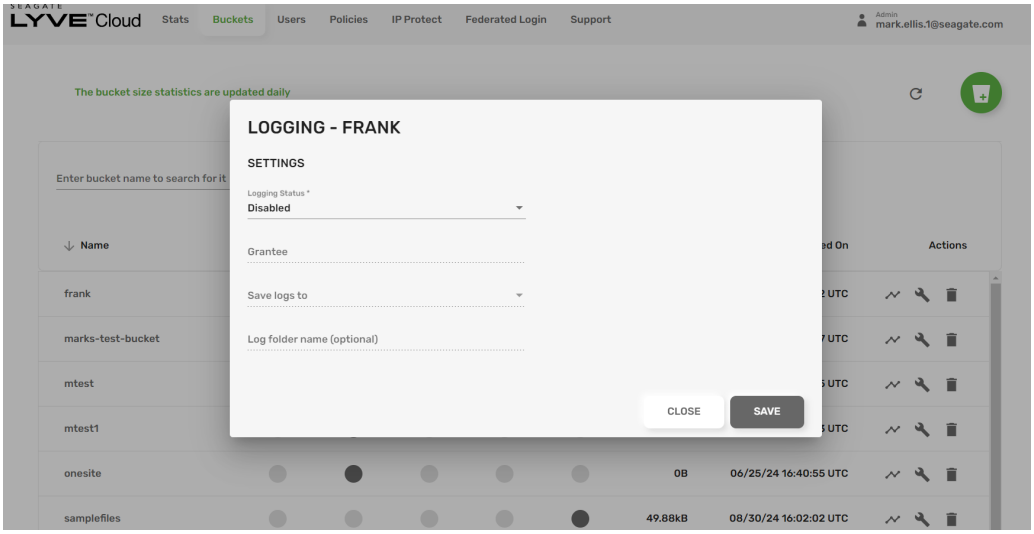
Processing time: Lifecycle processing runs in the background. Allow several days for changes to begin appearing, especially when the policy affects a large number of objects. In some cases—such as buckets with millions of objects—cleanup may take a week or more. If no changes are visible after two weeks, contact our support team.

If you need help applying lifecycle rules to only part of a bucket or want tailored advice for complex data-removal scenarios, our support team can assist with appropriate strategies.

S3 audit/bucket logs

Server access logging provides detailed records for the requests that are made to a bucket. Logs are useful for many reasons, such as security and usage checks. By default, the system does not provide access logs—you must explicitly enable this feature. When you enable bucket logging, the system will write all the actions on the monitored bucket to a destination that you choose. This provides a log of every action which occurs in a bucket.

Bucket logging is set up on the main bucket screen of the Lyve Cloud console via a button on the bucket summary line. This allows control of where the logs are generated, and their naming as shown below.



The following items should be noted around the bucket logging functionality:

- Bucket logs are written periodically by the system. The logging is not instant—it can take some time for a log to be written. Each log will likely contain multiple events from over a time-period. The frequency of log file creation, and time period which a log file covers, may change depending on multiple factors, including the number of events occurring and system processes generally.
- Bucket logging is done on a ‘best effort’ basis. All possible actions are taken to ensure the logs are complete—however, in some circumstances, it's possible that bucket logs may not be a complete record of every event which occurs on a bucket. It's also possible that duplicate records may be created in the logs. Although log records are rarely lost or duplicated, you should be aware that bucket logging is not guaranteed to be a complete accounting of all requests.
- It's strongly advised that you avoid writing bucket logs to the monitored bucket itself. This will

generate an infinite loop of events and will likely be counterproductive for any analysis of the traffic in the bucket. It also increases storage costs for the logs.
- Bucket logging will generate extra storage in the account, which will be charged at the normal rates.

Bucket log object names are generated in the following format:

bucket_logging_<endpoint>_<account>_<bucket>_<year>_<month>_<day>_<hour>_<minute>_<seconds>.<milliseconds>

The date/time is when the object was created.

The bucket logs events are in the following format:

```
0QTV8D9VN4P3N4CPV4HX24F2XZ mtest [08/Nov/2024:15:42:29 +0000] "134.204.180.68" "STX07YT7MIZNIDQHHQV49OSS" "ee3d7cc8fdfa3c0e69ac2df2a9ae99d8" s3:GetBucketVersioning "" "GET /mtest?versioning" "200" "-" "0" "0" 11 "0" "https://console.sv15.lyve.seagate.com/" "Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/130.0.0.0 Safari/537.36" "-" - SigV4 SSL AuthHeader - "-" 0QTV8D9VN4P3N4CPV4HX24F2XZ mtest [08/Nov/2024:15:42:29 +0000] "134.204.180.68" "STX07YT7MIZNIDQHHQV49OSS" "6db6f3490bfeb4262ba30f7978a06dfd" s3:GetBucketLogging "" "GET /mtest?logging" "200" "-" "0" "0" 6 "0" "https://console.sv15.lyve.seagate.com/" "Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/130.0.0.0 Safari/537.36" "-" - SigV4 SSL AuthHeader - "-"
```

The following table lists the descriptions of fields used in the sample log message above:

Field	Description	Example (from sample log above)
Bucket owner	An ID of the bucket that stores the object being copied. This is an internal reference item and included to ensure that the bucket log format matches the AWS format of bucket logs.	0QTV8D9VN4P3N4CPV4HX24F2XZ
Bucket	The name of the bucket that stores the object that's being copied.	mtest

Time	The time at which the request was received. These dates and times are in Coordinated Universal Time (UTC). The format, using strftime() terminology, is [%d/%B/%Y:%H:%M:%S %z].	[08/Nov/2024:15:42:29 +0000]
Remote IP	The apparent IP address of the requester. Intermediate proxies and firewalls might obscure the actual IP address of the machine that's making the request.	"134.204.180.68"
Requester	The access key ID of the requester, or a hyphen - for unauthenticated requests.	"STX07YT7MIZNIDQHHQV49OSS"
Request ID	A string generated by the system to uniquely identify each request.	"ee3d7cc8fdfa3c0e69ac2df2a9ae99d8"
Operations	The operation which was requested to be performed.	s3:GetBucketVersioning
Key	The key (object name) of the object being copied, or "" if the operation doesn't take a key parameter.	""
Request-URI	The Request-URI part of the HTTP request message. In sample log: "GET /mtest?versioning".	"GET /mtest?versioning"

HTTP status	The numeric HTTP status code of the GET portion of the copy operation. Example: `200`.	`200`
Error code	The S3 Error code of the GET portion of the copy operation, or `""` if no error occurred.	`""`
Bytes sent	The number of response bytes sent, excluding the HTTP protocol overhead. This can be `0`.	`0`
Object size	The total size of the object in question. This can be `0`.	`0`
Total time	The number of milliseconds that the request was in flight from the server's perspective. This value is measured from the time that your request is received to the time that the last byte of the response is sent. Measurements made from the client's perspective might be longer because of network latency.	`11`

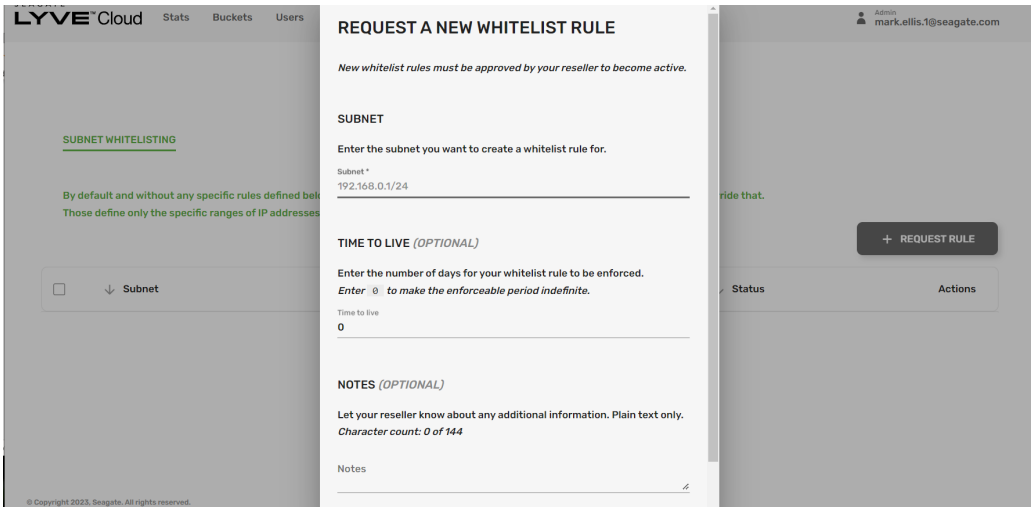
Turn-around time	The number of milliseconds that the system spent processing your request. This value is measured from the time that the last byte of your request was received until the time that the first byte of the response was sent. The value can be `0` if the response was instantly actioned.	`0`
Referrer	The value of the HTTP Referrer header, if present. HTTP user-agents (for example, browsers) typically set this header to the URL of the linking or embedding page when making a request.	`"https://console.sv15.lyve.seagate.com/"`
User-agent	The value of the HTTP User-Agent header.	`"Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/130.0.0.0 Safari/537.36"`
Version Id	The version ID of the object being copied, or `"-"` if the x-amz-copy-source header didn't specify a version or that is not relevant for the operation being performed.	`"-"`
Host Id	The S3 extended request ID, if valid.	`-`
Signature version	The signature version, SigV2 or SigV4, that was used to authenticate the request, or `-` for unauthenticated requests.	`SigV4`

Cipher suite	The Secure Sockets Layer (SSL) cipher that was negotiated for an HTTPS request, or `-` for HTTP.	`SSL`
Authentication type	The endpoint that was used to connect to Lyve object cloud. If it is from an internal system, the value will be `-`.	`-`
Host header	The time at which the request was received. These dates and times are in Coordinated Universal Time (UTC). The format, using strftime() terminology, is [%d/%B/%Y:%H:%M:%S %z].	`[08/Nov/2024:15:42:29 +0000]`
TLS version	The Transport Layer Security (TLS) version negotiated by the client. The value is one of following: `TLSv1.1`, `TLSv1.2`, `TLSv1.3`, or `""` if TLS wasn't used.	`""`

IP source control

Lyve Cloud Object Storage provides a way of limiting the IP connections to an account (to both the console and S3 REST calls). By default, the system allows access from any IP range – which is the equivalent of having a default setting 0.0.0.0/0 mask setting.

IP source control can be set using custom API REST calls or the Lyve Cloud console, for example:



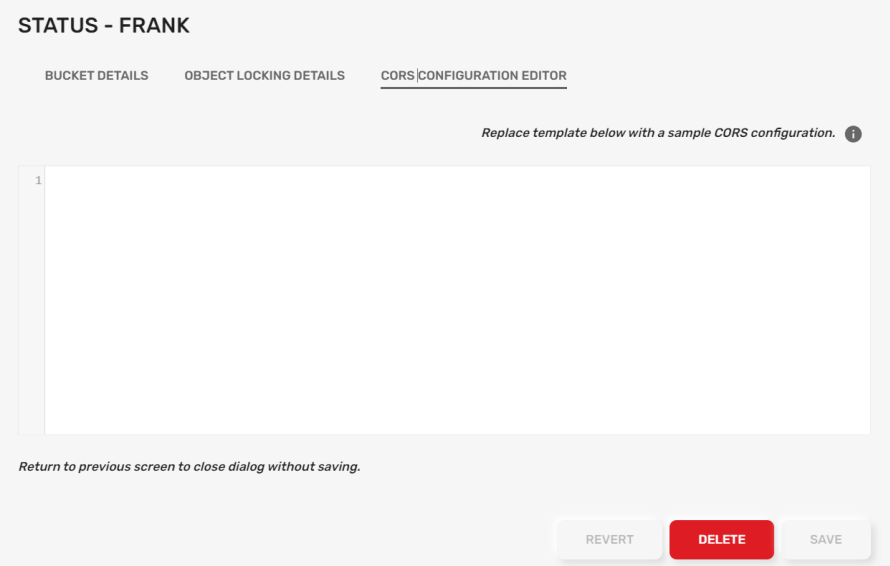
If any IP masks are specified in the console's IP Protect page or through S3 REST calls, these will limit access to those source IP's. Multiple rule sets are allowed, and traffic is allowed if the IP source address matches any one of the allowed subnet masks. So, when a rule is specified then those override the default value. Any standard IP V4 mask is allowed, and the system support an unlimited number of sub-net masks.

If a mistake is made and you lock yourself out of your own account, contact your reseller or Seagate support to help restore service.

CORS support

Cross-origin resource sharing (CORS) defines a way for client web applications that are loaded in one domain to interact with resources in a different domain. With CORS support, you can build rich client-side web applications which use our S3 solution for data storage and selectively allow cross-origin access to your resources.

The system supports CORS, which is defined by AWS. CORS support is configured via the S3 commands or via the bucket settings in the Lyve Cloud console (see below). All the standard CORS policies and controls are supported at this tme.



Storage Class support

If you have purchased Infrequent Access storage (see the Seagate sales team for details), the system supports the Storage Class settings of STANDARD and STANDARD_IA. Setting the Storage Class to STANDARD_IA tells Lyve Cloud Object Storage that this object is less likely to be accessed in the future. By default, objects are set to STANDARD level.

The Storage Class of an object is defined upon creation. When adding the object via the object creation REST command, you can set the Storage class to STORAGE_IA by adding thx-amc-storage-class: STANDARD_IA parameter. To change the setting of an existing object, you must recreate the object with the new Storage Class value, using either a PUT or COPY command. The Storage Class for existing objects can be seen on a list of the object.

Setting the Storage class to STANDARD_IA allows the system to mange the storage of the object differently. However, all features and functions are available for those objects. They are still accessible via the standard means. The number and frequency of interactions with objects at the STANDARD_IA level may be more restricted by the service.

Connect S3 Clients

You can configure third-party clients including AWS CLI, Cyberduck, S3 Browser, Mountain Duck, or Rclone to manage data. Lyve Cloud Object Storage is an S3-compatible storage service for data-intensive applications such as data backup and analytic workloads, that leverage multi-petabyte data lakes. You can also use any other compatible third-party client to copy and move files, manage files and folders, and synchronize folders between Lyve Cloud Object Storage and your local storage, once you've established a connection.

See the following chapters:

- Connect Using AWS CLI
- Connect Using Cyberduck
- Connect Using S3 Browser
- Connect Using Mountain Duck
- Connect Using Rclone

Connect Using AWS CLI

For command line interface (CLI) access to the AWS S3 system, Lyve Cloud recommends the industry standard AWS interface: https://docs.aws.amazon.com/cli/latest/userguide/getting-started-install.html

Before configuring, you'll need the following:

- Username
- Access Key ID
- Secret Access Key

> **Note**—In the examples that follow, system prompts and responses are shown in black, and users commands/entries are shown in red.

After installing the CLI, configure the CLI for your access. Use the Access Key ID/Secret Access Key to set that in the interface:

```
% aws configure
AWS Access Key ID [None]: <Enter your Access Key ID>
AWS Secret Access Key [None]: <Enter your Secret Access Key ID>
Default region name [a]:
Default output format [a]:
%
```

For example:

```
% aws configure
AWS Access Key ID [None]: RSTOR1969ZYBQ7UR6PWU92GVZC
AWS Secret Access Key [None]: 96d3k4z01olnH/Cxu5tB7zD/9KY7s4t2pOnY4Wjg6QO
Default region name [a]:
Default output format [a]:
%
```

You can now use normal S3 commands. Note that the 'endpoint-url' parameter must be included in any requests using this interface to ensure that the command gets routed to the Lyve Cloud Object Storage instance, and not the default AWS cloud.

An example of a simple search request at the top level of access is shown below. The response shows only

one bucket which the user has access to, which is called 'test':

```
% aws s3 ls --endpoint-url <your-endpoint-url>
1.      3:24:31 test
%
```

For a user to search that bucket, the command would be:

```
% aws s3 ls s3://test/ --endpoint-url <your-endpoint-url>
2020-03-11 13:26:22     411923 dog_bed_after.jpg
2020-03-11 13:25:44     434088 dog_bed_before.jpg
%
```

In the example above, there were two files in the test bucket.

The system can be used to validate access or test functions/features at the CLI level. The AWS CLI help function provides details of the other commands and functions.

Connect Using Cyberduck

Use Cyberduck to connect with Lyve Cloud Object Storage and transfer your data. For more information, review the Cyberduck Tutorial and Cyberduck Quick Reference Guide.

Prerequisites

- Download and Install Cyberduck.
- Register the S3 (HTTPS) profile for preconfigured settings. For more information, see Generic S3 profiles.

To enable and register the S3 (HTTPS) profile:

1. Select **Edit**, and then select **Preferences**. Mac users: Select **Cyberduck**, and then select **Preferences**.
2. In the **Profiles** tab, select **S3 (HTTPS)** from the connection profiles list.

Alternatively, to register the S3 (HTTPS) profile:

1. Open the S3 (HTTPS) profile. Copy the file contents into a notepad/any text editor.
2. Save the notepad file name with the .cyberduckprofile extension, and change the 'Save as' type to **All Files**.

You need both the Access Key ID and Secret Access Key for each account you plan to connect with Cyberduck. For more information, see Creating your S3 Access key.

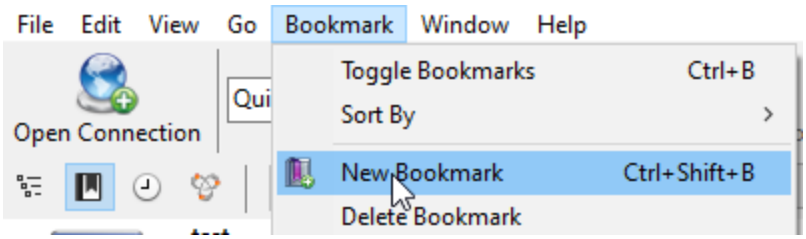
> **Note**—Consult your organization's policies and the EULA policies of the software before downloading 3rd party applications.

Connect Cyberduck to Lyve Cloud Object Storage

Bookmarks store the details of the connection to easily re-connect to the server.

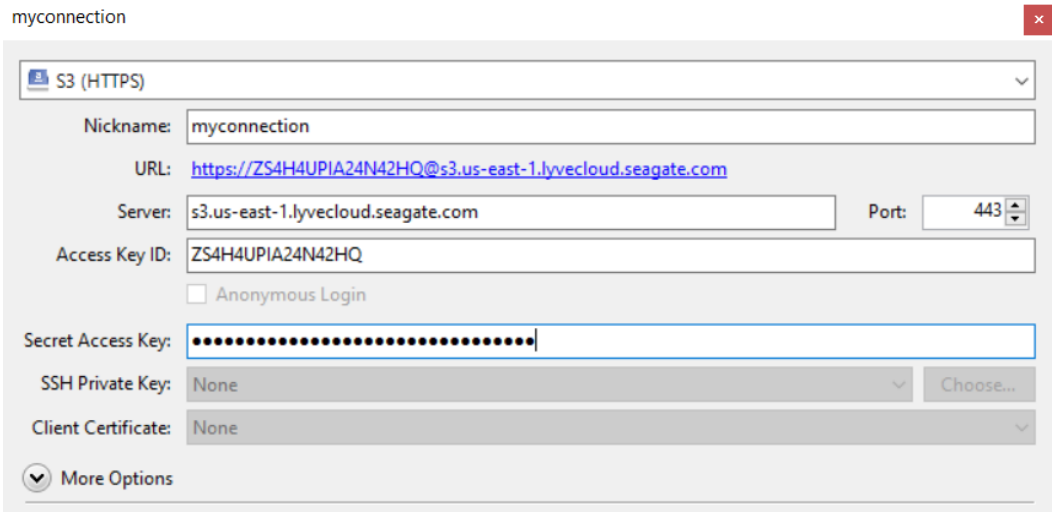
To connect Cyberduck to Lyve Cloud Object Storage:

1. In Cyberduck, select **Bookmark|New Bookmark**. Mac users: Select **+** in the bottom left to add a new bookmark.

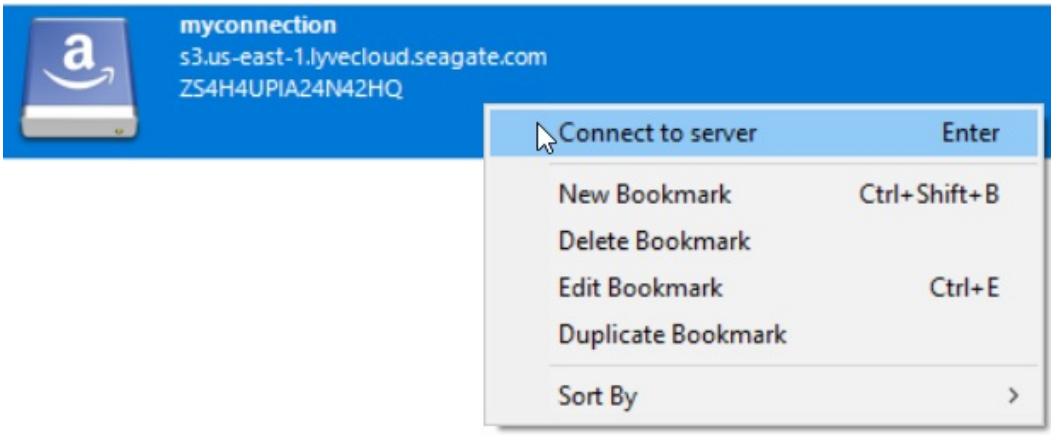


2. Select the **S3 (HTTPS)** protocol from the list.
3. Enter the following mandatory details to add your connection to Lyve Cloud Object Storage:

Field name	Description
Nickname	Enter a name for the bookmark.
URL	Displays the URL once you enter the server and access key in the following format: http://
Server	Enter your Lyve Cloud Object Storage S3 endpoint. Lyve Cloud Object Storage supports region-specific S3 endpoints. To access buckets created in different regions in the S3 client, add an endpoint connection for each of the regions.
Port	Enter `443` as the port number to access the server.
Access Key ID	Enter your Access Key ID, a private key for authentication to connect a bucket created in Lyve Cloud Object Storage . The access key is displayed when you create a key for use of the service.
Secret Key Access	Enter your secret key, a private key password for authentication to connect a bucket created in Lyve Cloud Object Storage . The secret key displays when you create a new service account in Lyve Cloud Object Storage . It is not displayed after that so retain/save the key information for this use.



4. The bookmark is displayed once you close the window. Right-click the bookmark and select **Connect to Server**.



5. Select **Continue** to establish the connection.

   - View the buckets available in the created bookmark once the connection is established.
   - On the Cyberduck client, the **Disconnect** button is displayed in the top right corner. A green dot appears to the right of active bookmarks, signifying an established connection. If no connection is established, the Disconnect icon is greyed out.

6. Right-click the bucket or select Actions to perform various operations or actions. For more information, see **Manage data** below.

Manage data

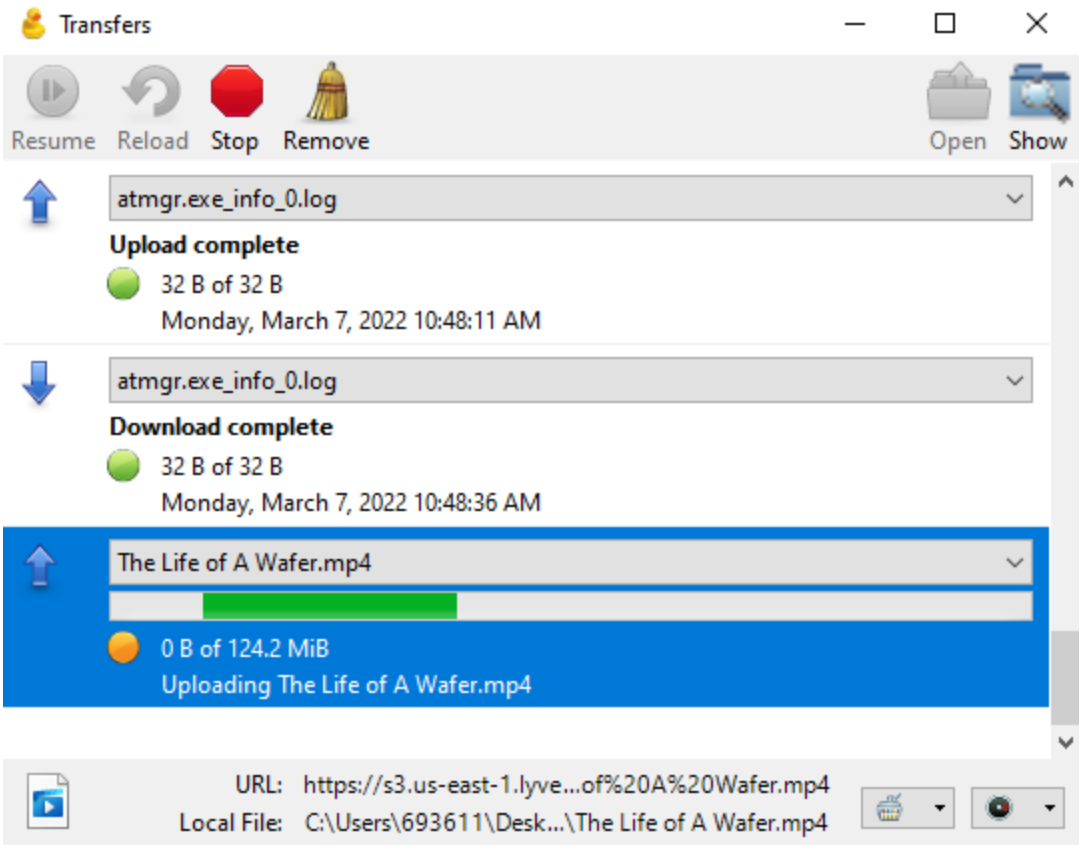
You can perform various actions once a connection is established between Cyberduck and Lyve Cloud Object Storage. For more information, see Cyberduck Help.

Upload data to a bucket

Note—The object name can contain special characters such as `@ # * $ % & ! ? , ; ' " | + = < > ^ ( ) { } [ ]`, as well as alphanumeric characters like 0-9, a-z, A-Z . However, using any of these characters may cause issues due to limiting factors of the S3 client SDK.

To upload data to a bucket:

1. Right-click a bucket and select **Upload.**
2. Select a file to upload. After the file transfer progress status is complete, you can view the file.



3. Accept the certification installation and select **Continue**.

Note—Certificate installation is prompted only when the certificate is installed for the first time.

Download data to local storage

To download data to local storage:

1. Expand the bucket where files are available.
2. Right-click the file to download and select one of the following options:

Download	Download a file to the predefined path. The Transfers dialog displays the connection status. In the Download dialog, select **Continue**. You can view the remote file location and the local file location, but you cannot change the download path. Once the download is complete, the Transfers dialog displays the status.
Download As	Download a file in the required format. Select **Save as Type** from the list, and then select **Save**.
Download To	Download the file to a specific location. Select the download folder in the Browse to Folder dialog, or create a new folder. Select **Save**.

Delete data from a bucket

To delete data from the bucket:

1. Expand the bucket from which to delete data.
2. Right-click the data file, and then select **Delete.**
3. In the Confirmation prompt, select **Delete**.

Create a new folder

To create a new folder:

1. Select **Select** and open a bucket.
2. Right-click inside the bucket and select **New Folder.**
3. Enter the folder name in the Create New Folder screen.

Delete a folder

To delete a folder:

1. Navigate to the bucket from which to delete the folder.
2. Right click the folder and select **Delete**.
3. In the confirmation prompt, select **Delete**.

Disconnect Cyberduck from Lyve Cloud Object Storage

To disconnect Cyberduck from Lyve Cloud Object Storage:

1. Open Cyberduck to view all the available bookmarks or connections.
2. Select **Disconnect** in the top-right corner of the Cyberduck client.

> **Note**—A green dot indicates an active bookmark and an established connection. However, if the connection is not established, the Disconnect icon is greyed out.

Connect Using S3 Browser

Use S3 Browser to connect with Lyve Cloud Object Storage and manage your data transfer. For more information on S3 Browser see,S3 Browser Help.

Prerequisites

You will need the access key and secret key for each account you'll be using to connect with S3 browser.

> **Note**—Consult your organization's policies and the EULA policies of the software before downloading 3rd party applications.

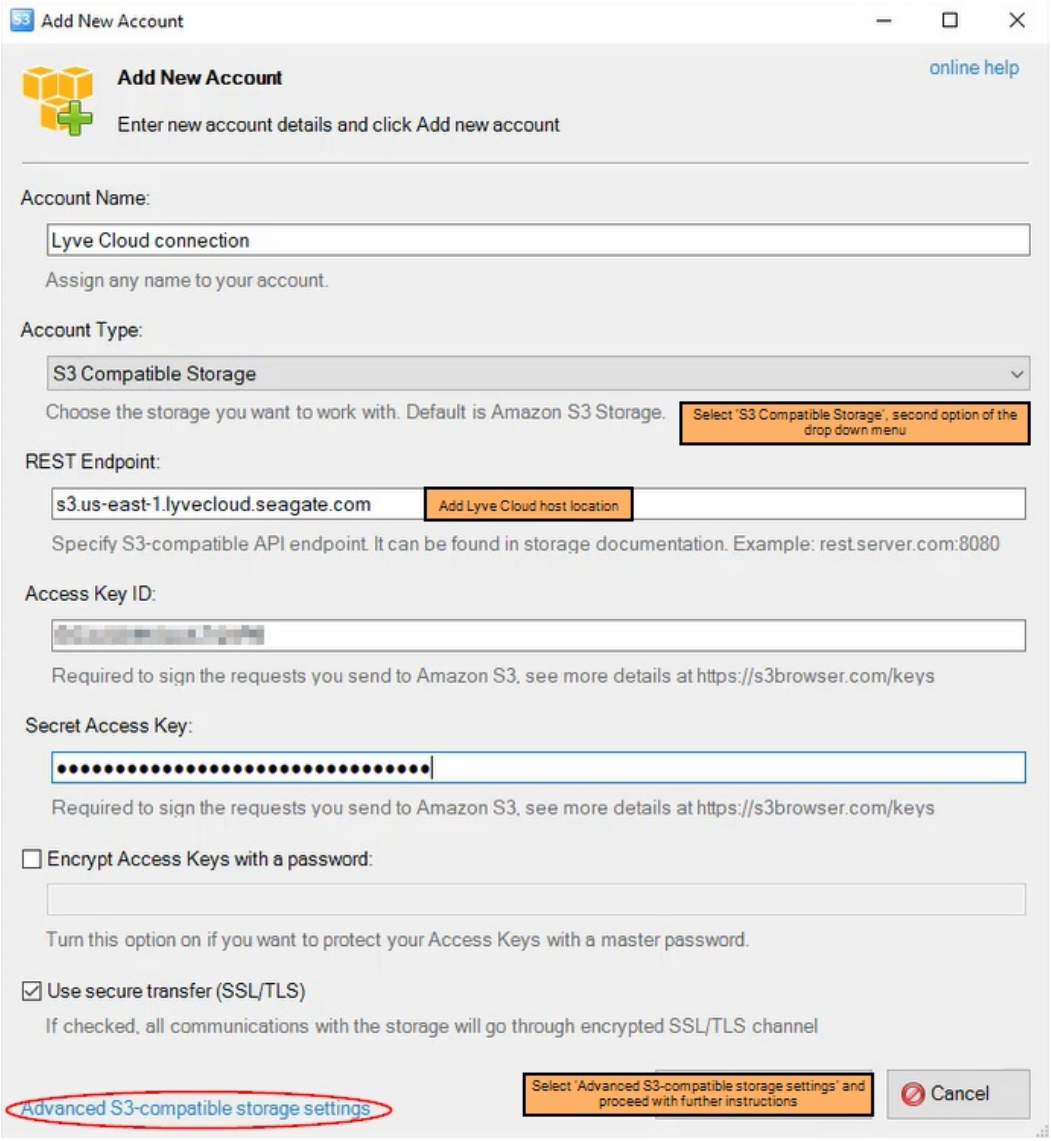
Connect S3 Browser to Lyve Cloud Object Storage

To connect S3 Browser to Lyve Cloud Object Storage:

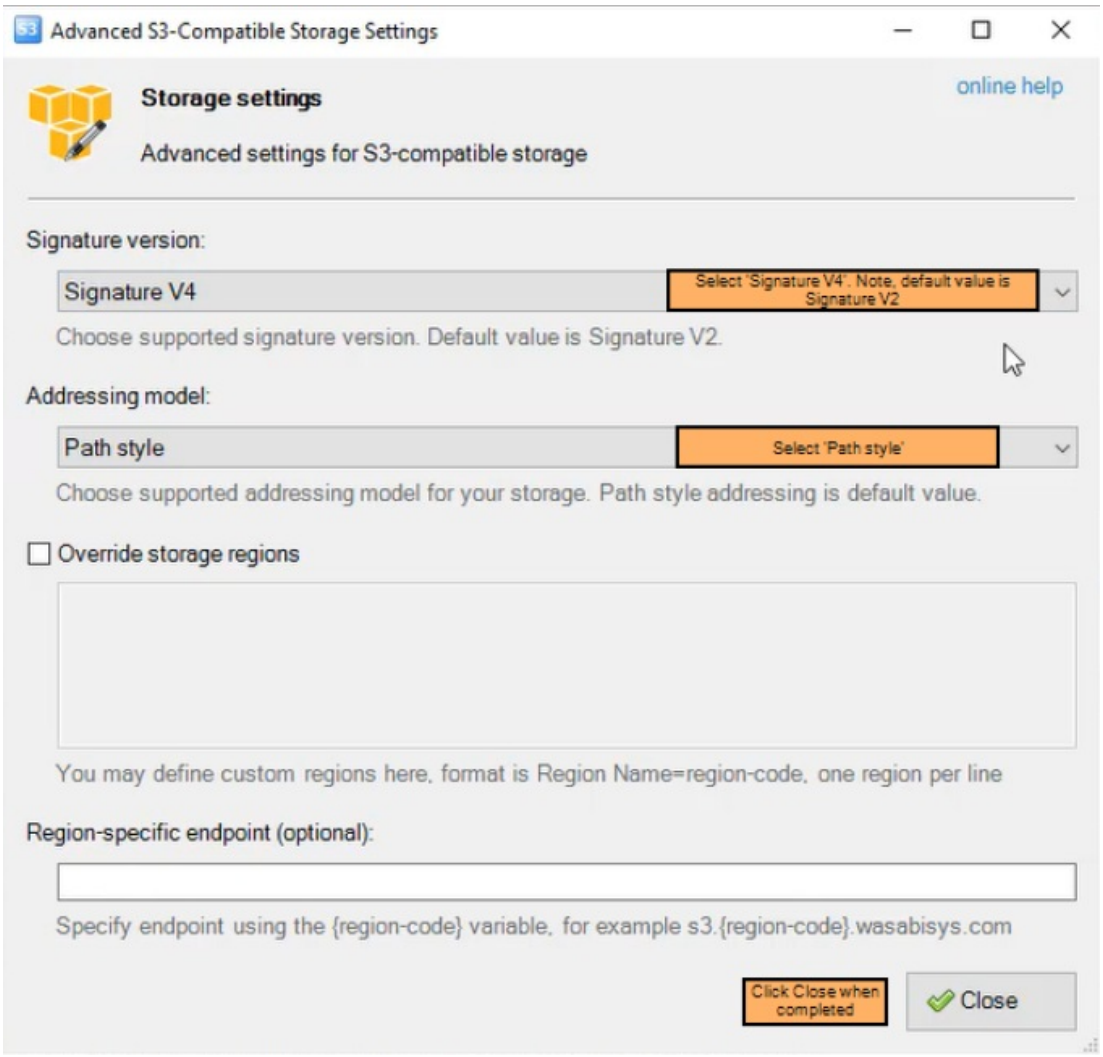
1. Open S3 Browser and select**Accounts**, then select**Add New Account**.
2. Enter the following mandatory details:

Field Name	Description
Account name	Enter an account name.
Account type	Enter the account type. Select**S3 Compatible Storage** from the list.
REST Endpoint	Enter your Lyve Cloud Object Storage S3 endpoint. Currently, Lyve Cloud Object Storage supports only region-specific S3 endpoints. To access buckets created in different regions in the S3 client, add an endpoint connection for each of the regions.
Access Key	Enter your access key. The access key displays when you create a new service account in Lyve Cloud Object Storage. A service account contains the bucket credentials for the Lyve Cloud Object Storage bucket.
Secret Key	Enter your secret key. The secret key displays when you create a new service account in Lyve Cloud Object Storage.

Field Name	Description
Use secure transfer (SSL/TLS)	Select this option to ensure all communication with the storage passes through encrypted SSL/TLS.
Advanced S3 Compatible storage settings	Select the signature version and addressing model in the advanced settings.



Advanced S3 compatible storage settings



Field Name	Description
Signature Version	Select **Signature V4**. For more information, see S3 Browser's help documentation.
Addressing Model	**Path Style** is selected by default and is the recommended setting. For more information, see S3 Browser's help documentation.

Once the connection is established, the bucket displays in the left pane. If there is no connection, an error message is displayed.

Manage data

Perform various actions once the connection between S3 Browser and Lyve Cloud Object Storage is established. For more information, see S3 Browser's help documentation.

Upload data to a bucket

> **Note**—The object name can contain special characters such as@ # * $ % & ! ? , ; ' " | + = < > ^ ( ) { } [ ], as well as alphanumeric characters such as 0-9, a-z, A-Z . However, using any of these characters may cause issues due to limiting factors of the S3 client SDK.

To upload data to a bucket:

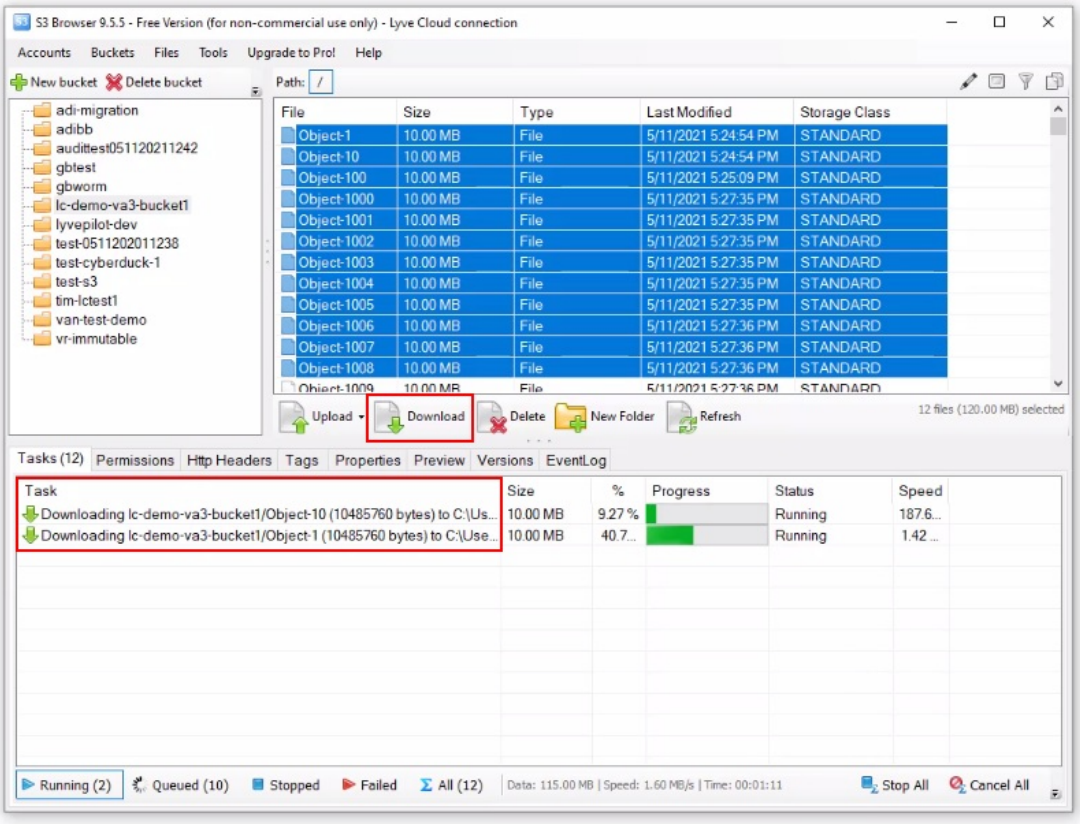
1. Select the bucket, and then select**Upload.**
2. Select **Upload file(s)** or **Upload folder(s)**.
3. Select the file, and then select**Open**.

Download data to local storage

To download data to your local machine:

1. Select the bucket where the data file is available.
2. Select the folder or file(s) to download, and then select**Download**.

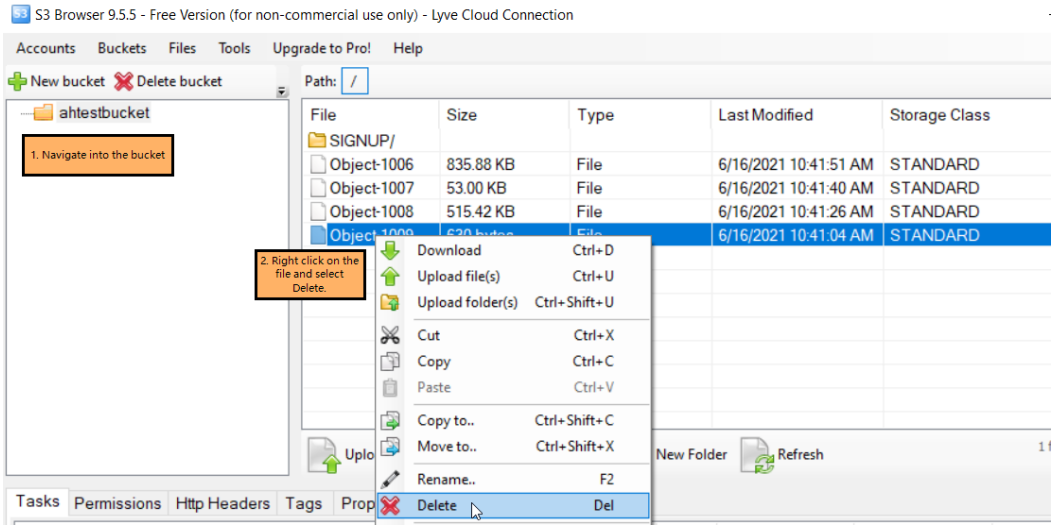
The Tasks tab displays the upload or download progress.



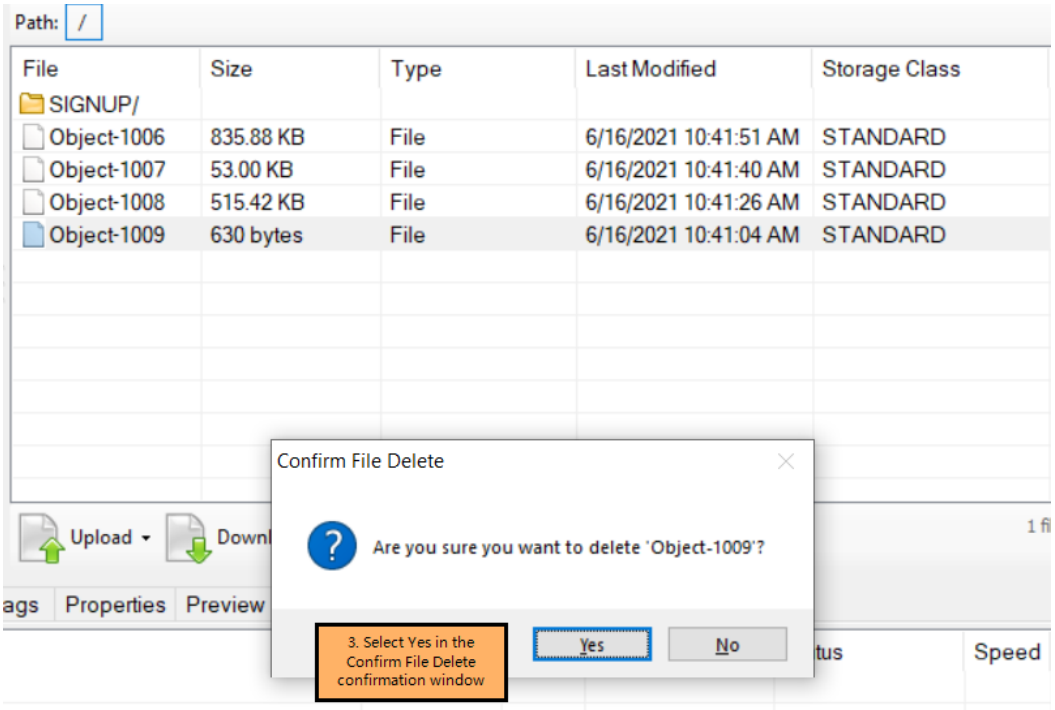
Delete data from a bucket

To delete data from a bucket:

1. Navigate into the bucket, select the file from the right pane, and select **Delete**.



2. Select **Yes** in the Confirm File Delete dialog.

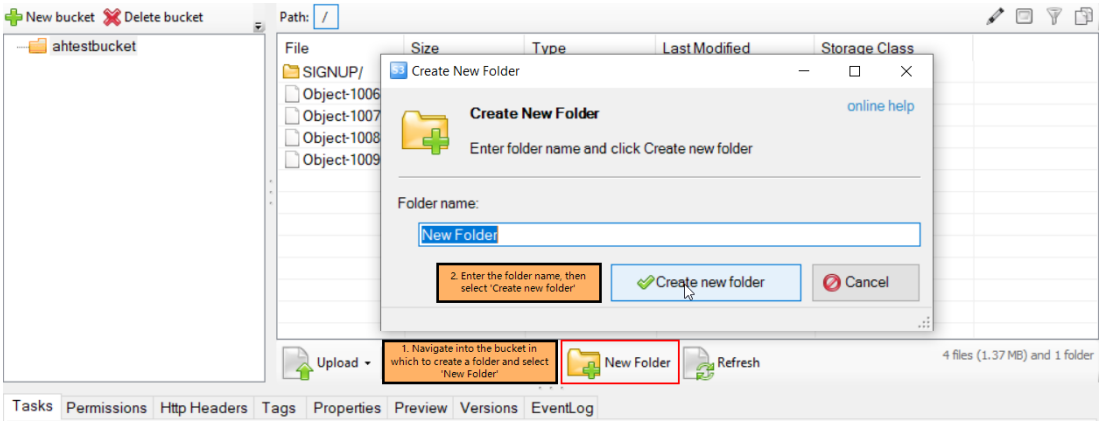


Create a new folder

To create a new folder:

1. Navigate into the bucket in which to create a folder.

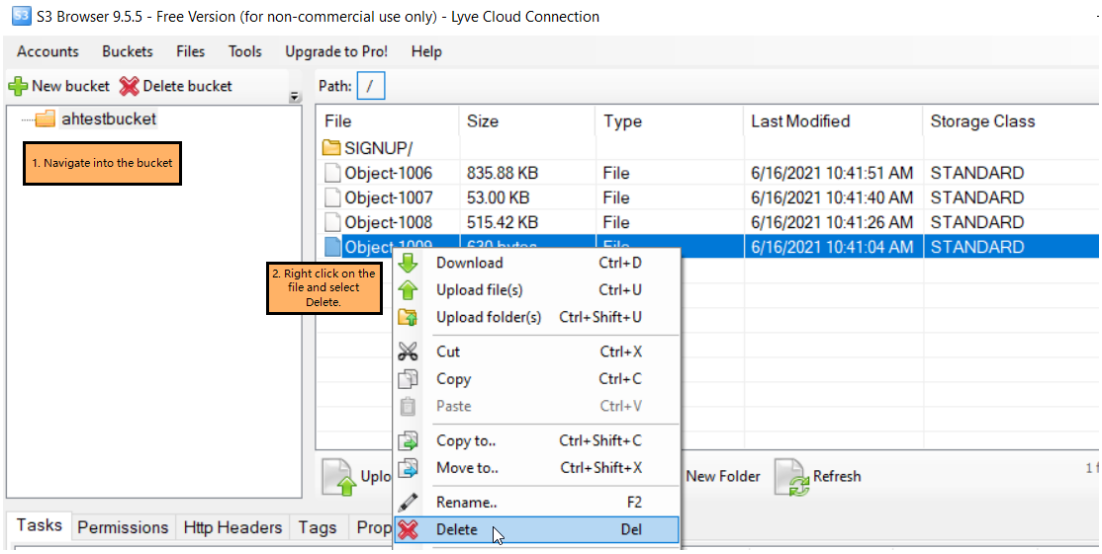
2. Select **New Folder**. Enter a folder name, and then select **Create New folder**.



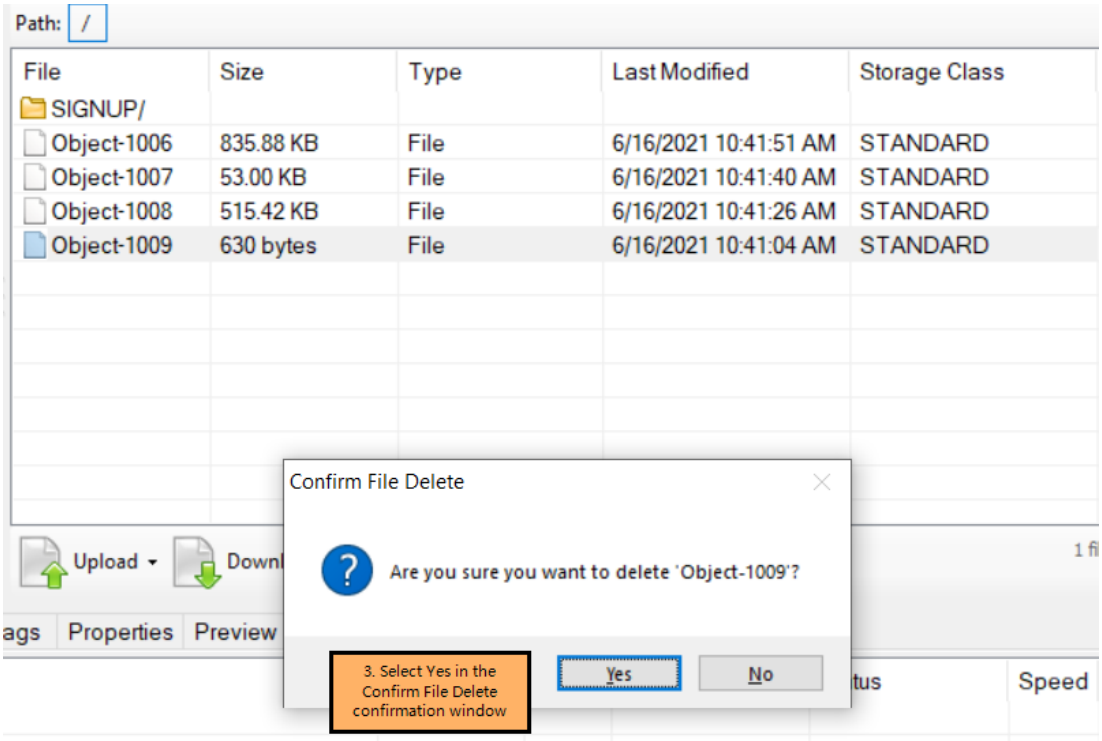
Delete a folder

To delete a folder:

1. Open the bucket, right-click the folder to delete, then select **Delete**.



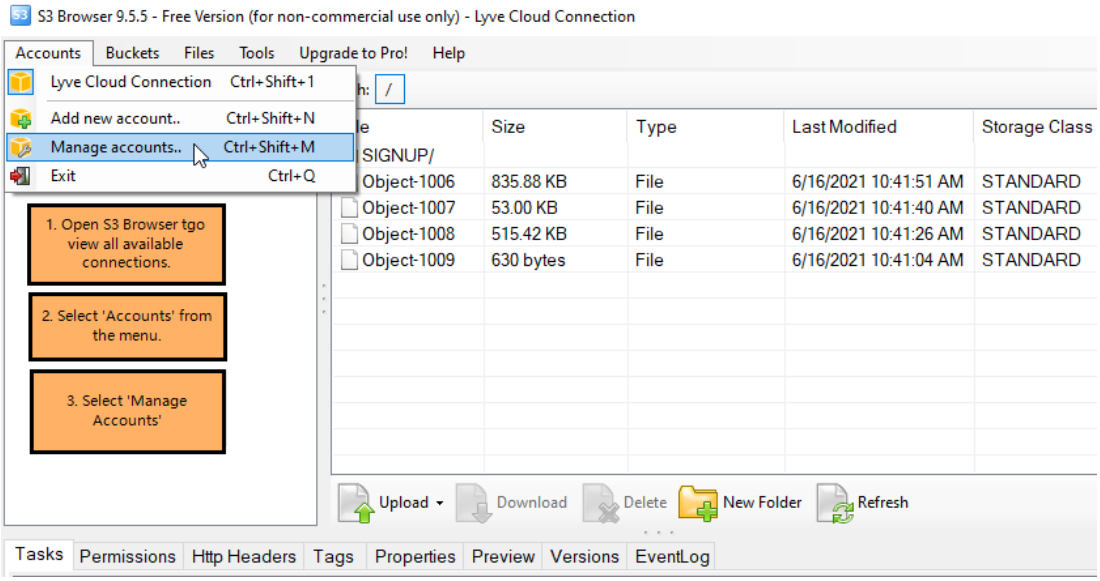
2. Select **Yes** in the Confirm File Delete dialog.



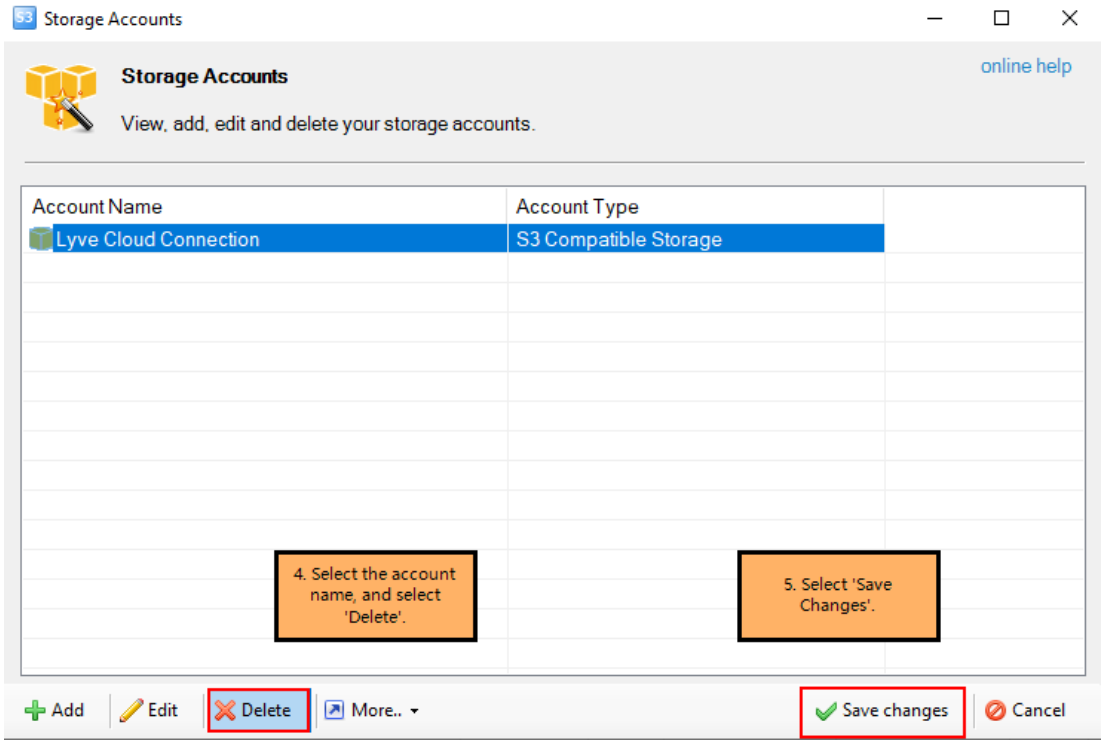
Disconnect S3 Browser from Lyve Cloud Object Storage

To disconnect S3 browser:

1. Open S3 Browser to view all available connections.
2. Select **Accounts** from the menu.
3. Select **Manage Accounts. S**elect the account name, and then select**Delete**.



4. Select **Save Changes**.


Storage Accounts
Storage Accounts
View, add, edit and delete your storage accounts.
online help
Account Name
Account Type
Lyve Cloud Connection
S3 Compatible Storage
4. Select the account name, and select 'Delete'.
5. Select 'Save Changes'.
Add
Edit
Delete
More..
Save changes
Cancel

Connect Using Mountain Duck

Use Mountain Duck to mount your Lyve Cloud Object Storage as a disk in the Windows File Explorer or Mac OS Finder, and manage your files through a familiar interface. For more information on Mountain Duck see, Mountain Duck Help.

Prerequisites

You will need the access key and secret key for each account you'll be using to connect with S3 Browser.

> **Note**—Consult your organization's policies and the EULA policies of the software before downloading 3rd party applications.

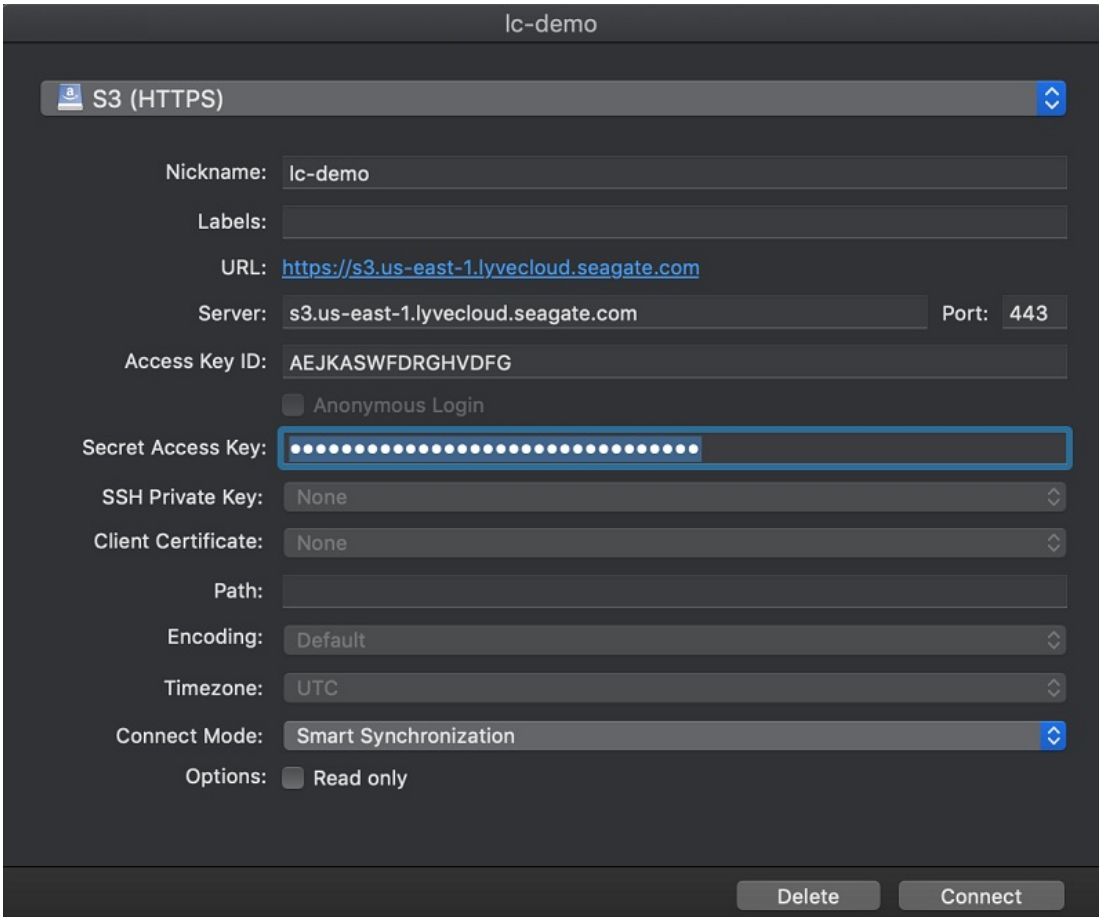
> **Note**—Differences between Windows and macOS are mentioned specifically. Othewise, instructions are similar for both operating systems.

Connect Mountain Duck to Lyve Cloud Object Storage

To connect Mountain Duck to Lyve Cloud Object Storage:

1. Download Mountain Duck's S3 (HTTPS) profile for preconfigured settings. For more information, see their Generic S3 profiles documentation.
2. Open the downloaded file with Mountain Duck. The New Connection dialog appears.

> **Note**—The dialog for macOS does not have a Drive Letter field.

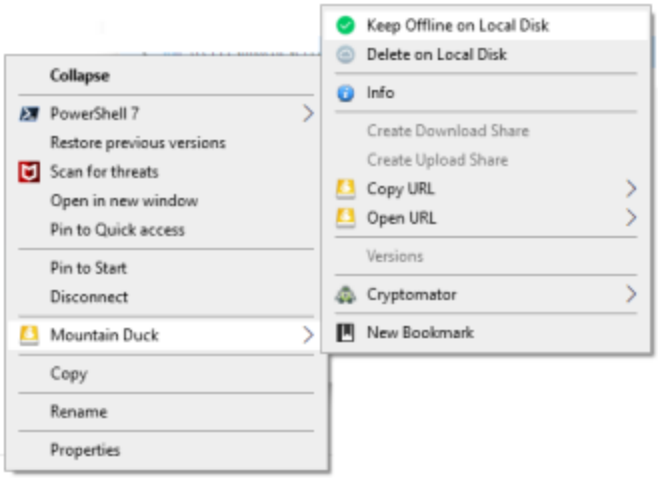


3. Enter the following information:

Field Name	Description
Nickname	Enter a unique name. This will be the name of your connection bookmark.
Server	Enter the endpoint based on the region. Currently, Lyve Cloud supports only region-specific S3 endpoints. To access buckets created in different regions in the S3 client, add an endpoint connection for each of the regions.
Port	This should populate as 443. If not, enter that port number.
Access Key ID	Enter your access key ID.
Secret Access Key	Enter your secret key.
Drive Letter(Windows only)	Enter a drive letter so that Mountain Duck always uses that same letter for the mounted drive.

4. Once the fields have been filled, select **OK**. This creates your connection bookmark.

5. Select the Mountain Duck icon in the Windows system tray or the macOS' menu bar.
6. Select your connection bookmark, and then select **Connect**. When the connection is created, a notification appears.
7. Select the new drive and right-click or Ctrl+click in the File Explorer or Finder to bring up the context menu. Right-click in the folder to get the context menu for macOS.
8. Select **Mountain Duck** and select **Keep offline on local disk** to sync all the data to local drive.



When connected, the drive and folder contents display their sync status. Look for a circle in the lower-left corner of the folder or file icon. Once mounted, all the files are stored on your local drive.

icon	Meaning
	In Progress. Synchronization is in progress for this item.
	In sync. This item is selected to be synced, and the content will always be available offline.
	Sync error. This item cannot be synchronized.
	Up to date. This item is synced and up to date.
	Ignored. The file is available in its temporary location and never synced to cloud or remote storage.
	Paused. The sync on that item is paused.
	Online only. This item is available in the cloud but can be opened and edited when you have an active connection to the server.

To learn more about various Mountain Duck options and sync modes, see Mountain Duck's Help

documentation.

Manage data

Once Lyve Cloud Object Storage is mounted as a drive, managing your files works much the same as working in any other network drive. Many of these operations may only be performed once a given bucket has synced with the local drive. Learn more about Mountain Duck's user interface documentation.

> **Note**—The object name can contain special characters such as @ # * $ % & ! ? , ; ' " | + = < > ^ ( ) { } [ ], as well as alphanumeric characters like 0-9, a-z, A-Z . However, using any of these characters may cause issues due to limiting factors of the S3 client SDK.

Copy data to a bucket

To copy data:

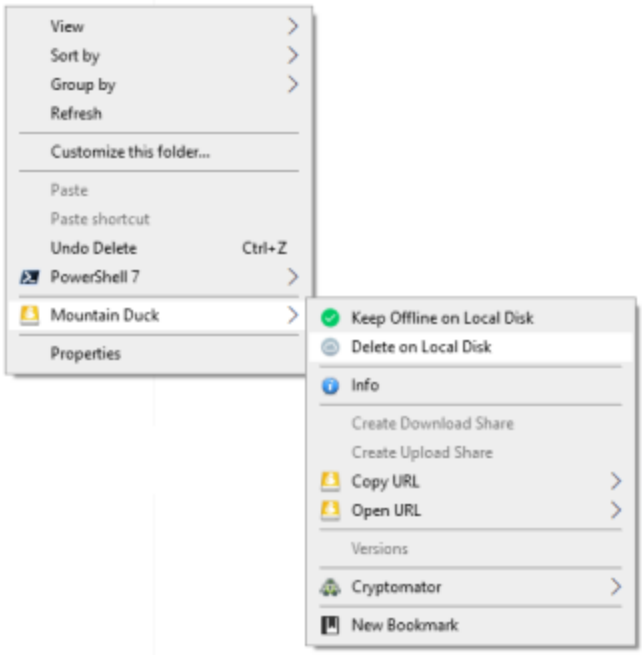
1. Select the data from your source drive and copy it.
2. Navigate to your destination and paste your data.

You can also drag and drop the data from one folder to another. If the bucket's or your service account's permissions do not allow you to write to that bucket, an error message appears.

Delete data from a bucket

To delete bucket data:

1. Navigate to the synced bucket from which you want to delete the data.
2. Right-click or Ctrl+click the file or folder inside the bucket to bring up the local context menu.
3. Select **Mountain Duck**, and then select **Delete on Local Disk**.



4. Right-click or Ctrl+click the file and select **Delete**, and then select **Yes** in the confirmation box to delete the object permanently from the bucket.

Create a folder in a bucket

To create a folder in a bucket:

1. Navigate into the bucket where you want to create a new folder.
2. Right-click or Ctrl+click the bucket and select **New Folder**.
3. Type a new folder name, and then select **Enter**.

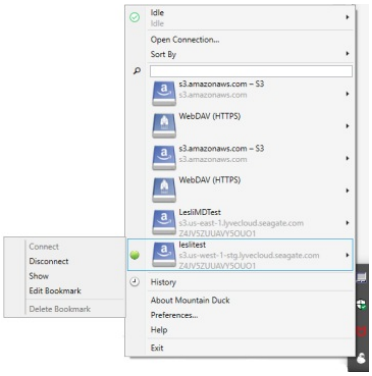
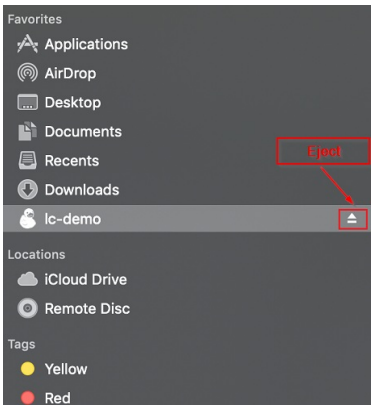
The new folder immediately begins syncing with its Lyve Cloud Object Storage destination.

Disconnect Mountain Duck from Lyve Cloud Object Storage

To disconnect Mountain Duck from Lyve Cloud Object Storage:

1. In the Windows system tray or the macOS menu bar, select the Mountain Duck client icon. The menu lists all of your connection bookmarks.
2. **Windows**: Select a connection bookmark, and select **Disconnect**. **macOS**: Select the Eject icon next to the connection bookmark, or click a connection and select **Eject**.

A notification pop-up appears when the connection is broken.

Windows	Mac
	

Idle
Idle
Open Connection...
Sort By
s3.amazonaws.com – S3
s3.amazonaws.com
WebDAV (HTTPS)
s3.amazonaws.com – S3
s3.amazonaws.com
WebDAV (HTTPS)
LesliMDTest
s3.us-east-1.lyvecloud.seagate.com
Z4IV5ZUUAVY5OUO1
leslitest
s3.us-west-1-stg.lyvecloud.seagate.com
Z4IV5ZUUAVY5OUO1
History
About Mountain Duck
Preferences...
Help
Exit
Connect
Disconnect
Show
Edit Bookmark
Delete Bookmark

Favorites
Applications
AirDrop
Desktop
Documents
Recents
Downloads
lc-demo
Eject
Locations
iCloud Drive
Remote Disc
Tags
Yellow
Red

Connect Using Rclone

Use Rclone to connect with Lyve Cloud Object Storage and manage your files from the command line or mount the cloud storage as a drive.

Prerequisites

You will need the access key and secret key for each account you'll be using to connect with Rclone.

> **Note**—Consult your organization's policies and the EULA policies of the software before downloading 3rd-party applications.

Connect to Lyve Cloud from Linux

Install Rclone

1. Download Rclone for Linux, then extract the rclone binary to your desired location.
2. To use Rclone, open a terminal window and navigate to the directory where you saved the executable.

Configure Rclone to connect to Lyve Cloud Object Storage

To configure a remote connection with Rclone:

1. Run **rclone config** to setup and select `n` for a new remote.

```
No remotes found - make a new one
n) New remote
s) Set configuration password
q) Quit config
n/s/q> n
```

2. Enter a name for the configuration.

```
name> <Name>
```

3. Select **s3** storage.

```
Type the storage to configure.
Choose a number from below, or type in your own value
1 /Fichier
 \(fichier)
2 /Akamai NetStorage
  \(netstorage)
3 / Alias for an existing remote
 \ (alias)
4 / Amazon Drive
 \ (amazon cloud drive)
5 / Amazon S3 Compliant Storage Providers including AWS, Alibaba, Ceph, Digital Ocean, Dreamhost, IBM COS,  Lyv
e Cloud, Minio,  RackCorp, SeaweedFS, and Tencent COS
 \ (s3)
6 / Backblaze B2
 \ (b2)  [snip]
46 / seafilehttp Connection
  \ (seafile)
 Storage> 5
```

4. Choose **Lyve Cloud** as the storage provider, or choose**Other**.

```
Choose the S3 provider.
Choose a number from below, or type in your own value
Press Enter for the default ("")
  1 / Amazon Web Services (AWS) S3
    \ (AWS)
 2 / Alibaba Cloud Object Storage System (OSS) formerly Aliyun
    \ (Alibaba)
 3 / Ceph Object Storage
    \ (Ceph)
 4 / Digital Ocean Spaces
    \ (DigitalOcean)
 5 / Dreamhost DreamObjects
    \ (Dreamhost)
 6 / IBM COS S3
    \ (IBMCOS)
 7 / Seagate Lyve Cloud
    \ (LyveCloud)
 8 / Minio Object Storage
    \ (Minio)
 9 / Netease Object Storage (NOS)
    \ (Netease)
10 / RackCorp Object Storage
    \ (RackCorp)
11 / Scaleway Object Storage
    \ (Scaleway)
12 / SeaweedFS S3
```

```
   \ (SeaweedFS)
13 / StackPath Object Storage
   \ (StackPath)
14 / Storj (S3 Compatible Gateway)
   \ (Storj)
15 / Tencent Cloud Object Storage (COS)
   \ (TencentCOS)
16 / Wasabi Object Storage
   \ (Wasabi)
17 / Any other S3 compatible provider
   \ (Other)
Provider>7
```

5. Enter **false** to enter your credentials.

```
Get AWS credentials from the runtime (environment variables or EC2/ECS meta data if no env vars).
Only applies if access_key_id and secret_access_key is blank.Enter a boolean value (true or false).
Please Enter for the default ("false").Choose a number from below, or type in your own value
   1 / Enter AWS credentials in the next step
    \ "false"
   2 / Get AWS credentials from the environment (env vars or IAM)
     \ "true"
   env_auth>false
```

6. Enter your **access key** and **secret key**.

```
AWS Access Key ID.
Leave blank for anonymous access or runtime credentials.
Enter a string value. Press Enter for the default ("")
    access_key_id> <access key>
AWS Secret Access Key (password)
Leave blank for anonymous access or runtime credentials.
Enter a string value. Press Enter for the default ("")
     secret_access_key> <secret key>
```

7. Leave the **region** blank.

```
Region to connect to.
Leave blank if you are using an S3 clone and you don't have a region.
Enter a string value. Press Enter for the default ("")
Choose a number from below, or type in your own value
1 / Use this if unsure.
  | Will use v4 signatures and an empty region.
  \ ()
2 / Use this only if v4 signatures don't work.
```

```
 | E.g. pre Jewel/v10 CEPH.
 \ "other-v2-signature"
region> <>
```

8. Specify the endpoint for Lyve Cloud Object Storage.
   - Lyve Cloud Object Storage currently supports region-specific S3 endpoints. To access buckets created in different regions in the S3 client, add an endpoint connection for each of the regions.
   - Specify the endpoint URL for your S3 service in the specific regions.

```
Endpoint for S3 API.
Required when using an S3 clone.
Choose a number from below, or type in your own value.
Press Enter to leave empty.
 1 / Seagate Lyve Cloud US East 1 (Virginia)
   \ (s3.us-east-1.lyvecloud.seagate.com)
 2 / Seagate Lyve Cloud US West 1 (California)
   \ (s3.us-west-1.lyvecloud.seagate.com)
 3 / Seagate Lyve Cloud AP Southeast 1 (Singapore)
   \ (s3.ap-southeast-1.lyvecloud.seagate.com)
endpoint> 1
```

9. Press **Enter** to skip the location constraint as there is no location constraint

```
Location constraint - must be set to match the Region.
Leave blank if not sure. Used when creating buckets only.
Enter a string value.
Press Enter for the default ("")location constraint>
```

10. Choose default ACL (private).

```
Canned ACL used when creating and or storing or copying objects.
This ACL is used for creating objects and if bucket_acl isn't set, for creating buckets too.
Note that this ACL is applied when server-side copying objects as S3
It doesn't copy the ACL from the source but rather writes a fresh one.
Enter a string value. Press Enter for the default ("")
Choose a number from below, or type in your own value
1 / Owner gets FULL_CONTROL.
  | No one else has access rights (default).
   \(private)
2 / Owner gets FULL_CONTROL.
  | The ALLUsers group gets READ access.
  \(public-read)
3 /Owner gets FULL_CONTROL.
  | The ALLUsers group gets READ and WRITE access.
[snip]
```

```
acl>1
```

11. Select **n** to save the default advanced configuration.

```
Edit advanced config? (y/n)
y) Yes
n) No (default)
y/n>n
```

12. Review the displayed configuration and accept to save the **remote** and then quit. The config file should look like this:

```
NAME]
    type = s3
    Provider = LyveCloud
    env_auth = false
    access_key_id = xxx
    secret_access_key = yyy
    region = us-west-1
    endpoint = s3.us-east-1.lyvecloud.seagate.com
    acl = private
```

13. Click **y** to confirm the configuration.

```
y) Yes this is OK (default)
e) Edit this remote
d) Delete this remote
y/e/d>y
```

14. Type **q** to quit the configuration, else select any of the following to edit, delete, rename, copy, Set configuration password.

```
Current remotes:

Name                 Type
====                 ====
ashrcl               s3

e) Edit existing remote
n) New remote
d) Delete remote
r) Rename remote
```

```
c) Copy remote
s) Set configuration password
q) Quit config
e/n/d/r/c/s/q>q
```

Mounting Lyve Cloud Object Storage as a drive

Prior to mounting Lyve Cloud Object Storage as a drive, connect and test the connection by viewing the contents of one or more buckets using the *rclone ls* command.

To mount Lyve Cloud Object Storage as a drive, use this command where:

- **remote** is the name of the remote server
- **path/to/files** is the exact path to the bucket(s)
- **path/to/local/mount** is the local directory:

```
rclone mount remote:path/to/files /path/to/local/mount
```

For more information, see Rclone's mount command documentation.

Manage data

Here are several of the more commonly-needed commands for viewing and managing your data from the command line. See the Rclone docs, including information on global flags, for additional information.

View information about your buckets and directories

There are three list commands with easily readable output available: *ls, lsd,* and *lsl*.

1. To list all data in a certain bucket, where **remote** is the name of the remote and **path** is the name of the bucket:

```
rclone ls remote:path [flags]
```

2. To list the directories in a certain remote and see the **total directory size, modification time, and number of objects** in the directories:

```
rclone lsd remote:path [flags]
```

Or

```
rclone lsd remote: [flags]
```

3. To list all objects in a certain remote and see **modification time, size and path** where path is the remote path beginning with the bucket name. Any of the filtering options can be applied to this command.

```
rclone lsl remote:path [flags]
```

4. Learn more about ls, lsd, and lsl.

Video: Lyve Cloud - How to use Rclone list commands

Seagate on Vimeo: Lyve Cloud - How to use Rclone list commands

Upload data to a bucket

> **Note**—The object name can contain special characters such as @ # * $ % & ! ? , ; ' " | + = < > ^ ( ) { } [ ], as well as alphanumeric characters like 0-9, a-z, A-Z . However, using any of these characters may cause issues due to limiting factors of the S3 client SDK.

1. To upload data into a bucket, use either of these commands:

```
rclone copy C:/path/to/filename remote:path [flags]
rclone copy filename remote:path
```

2. You can also use *rclone copy* to copy a file or directory to a new location and rename the directory at the same time. Neither of these commands deletes the file from the source, and neither of these commands will copy unchanged files.
3. Learn more about copy and copyto.

Video: Lyve Cloud - How to use Rclone Copy-to and Copy-sync Commands

Seagate on Vimeo: Lyve Cloud - How to use Rclone Copy-to and Copy-sync Commands

Download data to local storage

This is the same as copying, but the source path is something in the remote or in a bucket, while the destination path is on your local storage.

```
rclone copy remote:path C:/path/to/filename [flags]
```

Delete data from a bucket

To delete files in a certain path from a certain bucket:

```
rclone delete remote:path [flags]
```

You can use flags to delete only files with certain characteristics:

For example, to delete files that are over 100MB:

```
rclone --min-size 100MB delete remote:path
```

To delete only a specific file:

```
rclone deletefile remote:path [flags]
```

Learn more about rclone delete and deletefile.

Create a new folder

To create a new folder, for example, a file named 'blue' in the current location:

```
rclone mkdir blue
```

Create folders in other paths, or with other permissions, by setting the properflags. Learn more about mkdir.

Delete a folder

To delete an empty folder, for example, a file named 'blue' in the current location:

```
rclone rmdir blue
```

Add flags to delete folders in locations other than the current directory, for example, an empty folder named 'blue' that contains other empty folders:

```
rclone rmdirs blue
```

The folders must be empty for `rmdir` or `rmdirs` to work.

Learn more about rmdir and rmdirs.

Copy data

To copy data:

1. First, connect rclone to Lyve Cloud. For step by step instructions, see Configure rclone to connect to Lyve Cloud Object Storage. Once the configuration is complete, the rclone.config file must be updated as:

```
[REMOTE NAME]type = s3provider = Otherenv_auth = falseaccess_key_id = XXXXXXXXXXsecret_access_key = YYYYYYYYYYYYYYYYYYYYYYYYYYendpoint = https://s3.us-east-1.lyvecloud.seagate.comacl = privateregion = us-east-1
```

2. Start copying data from the existing cloud provider buckets to the buckets created in Lyve Cloud. To copy data:

```
$ rclone copy  SOURCE REMOTE:[SOURCE BUCKET] <TARGET REMOTE>[TARGET BUCKET>/<PREFIX>]
```

   A. To copy all the data, including prefixes, from the source bucket to the target bucket:

```
$ rclone copy  SRT:[ SB ]  TRT:[TB]
```

B. To copy all objects with a prefix to the target:

```
$ rclone copy  SRT:[ SB ]/mypath1  TRT:[TB]/mypath1
```

Deleting a remote from Lyve Cloud

To delete a remote from Lyve Cloud, delete the remote's name using this command, changing REMOTE_NAME to the name of the remote to disconnect:

```
rclone config delete REMOTE_NAME [flags]
```

Learn more about `config delete`.

> **Note**—Rclone will also disconnect whenever you shut down your computer.

Migrate data

To migrate data:

1. Set the Source and the Target as remote. For more information, see Configure rclone to connect to Lyve Cloud Object Storage.
2. Sync the Source and Target remote using rclone sync command.

```
rclone sync <source remote name>:path <target remote name>:path
```

3. Once the source and the target are synced, all the data from the source is copied, removed or migrated to the target remotely.
4. You can use the following flags in the command to check the status of the sync/copy/migration.

- `--progress` Displays the real-time transfer progress.
- `--interactive`: Enables interactive mode and displays interactive for every action taken.

For more information on RClone, see https://rclone.org/s3/.

Video: Lyve Cloud - Use Rclone Delete and Purge commands

Seagate on Vimeo: Lyve Cloud - Use Rclone Delete and Purge commands

Frequently Asked Questions

General

What is Lyve Cloud?
How do I sign up for Lyve Cloud?
What are the S3 service URL's for Lyve Cloud‘s regions?
Is Lyve Cloud S3 compatible?
What kind of workloads or applications are ideal for Lyve Cloud services?
Can I store my data in Lyve Cloud and leverage resources and applications from other cloud providers?
How will I be notified of Lyve Cloud service updates?
Can I change my registered email address?
Can I reset my password?
How does Lyve Cloud assure data is consistent and all applications have a single source of truth?
How does Lyve Cloud manage data integrity and protect against data corruption?
Does Lyve Cloud guarantee permanent deletion of data?
Does Lyve Cloud support multipart uploads?
What level of uptime does Lyve Cloud support?
Are there a minimum and maximum sizes for stored objects?
What kind of data can I store in Lyve Cloud?
What versions of TLS / SSL does Lyve Cloud support?
Does Lyve Cloud support WORM/Write-Once-Read-Many for data immutability?
Does Lyve Cloud support management and data access auditing?
Can I connect Lyve Cloud with third-party tools?
Which operating systems environments are supported by the Lyve Cloud console?
Does Lyve Cloud support strong consistency?

Q:	What is Lyve Cloud?
A:	Lyve Cloud is a simple, trusted, and efficient on-demand solution for mass capacity storage. Predictable economics, verifiable trust, and ease of use make Lyve Cloud the right choice for storing your data.
Q:	How do I sign up for Lyve Cloud?
A:	To sign up for Lyve Cloud services, contact our sales team at sales.lyvecloud@seagate.com.
Q:	What are the S3 service URL's for Lyve Cloud‘s regions?

A:	It depends on the endpoint you have. The S3 service URLs for Lyve Cloud will be in the following locations:
	• Eastern region (Northern Virginia) is
	`https://s3.us-east-1.<endpoint>`
	• Western region (California) is
	`https://s3.us-west-1.<endpoint>`
	• Asia Pacific region (Singapore) is
	`https://s3.ap-southeast-1.<endpoint>`
	• Asia Pacific region (Tokyo) is
	`https://s3.ap-northeast-1.<endpoint>`
	• Central region (Oklahoma) is
	`https://s3.us-central-1.<endpoint>`
	• Central region (Texas) is
	`https://s3.us-central-2.<endpoint>`
	• Europe region (Frankfurt) is
	`https://s3.eu-central-1.<endpoint>`
	• Europe region (London) is
	`https://s3.eu-west-1.<endpoint>`
	
	The management console URL is
	`https://console.<endpoint>`

Q:	Is Lyve Cloud S3 compatible?
A:	Yes, Lyve cloud is S3-compatible object storage that uses S3 API, and it works with other S3-compatible applications.
Q:	What kind of workloads or applications are ideal for Lyve Cloud services?
A:	Lyve Cloud is designed to handle all of your data storage needs for workloads such as backup, disaster recovery and big data analytics.

Q:	Can I store my data in Lyve Cloud and leverage resources and applications from other cloud providers?
A:	Yes. Lyve Cloud is located close to Azure, Amazon, and other cloud providers to facilitate high-speed data transfer. Lyve Cloud lets you keep control of your data while leveraging computing and application resources from other cloud providers.
Q:	How will I be notified of Lyve Cloud service updates?
A:	The Lyve Cloud support team sends notifications regarding any system updates. Updates are sent to the email addresses of all registered Lyve Cloud users.
Q:	Can I change my registered email address?
A:	Currently, you cannot change the registered email address from within your Lyve Cloud account. For more information, contact the Lyve Cloud support team at support.lyvecloud@seagate.com.
Q:	Can I reset my password?
A:	Yes, you can reset your password. Select the Forgot Password link on the login screen and follow the instructions. You can also contact any of your Lyve Cloud admins to reset the password.
Q:	How does Lyve Cloud assure data is consistent and all applications have a single source of truth?
A:	Lyve Cloud uses a robust data consistency model for writing an object to the disk. Therefore, the data read by any application is identical, providing a single source of truth.
Q:	How does Lyve Cloud manage data integrity and protect against data corruption?
A:	Lyve Cloud maintains data integrity as a core mission. At the physical layer, Lyve Cloud addresses data integrity by using Seagate's enterprise-class hard disks and cloud-scale data durability and self-healing technologies. The configuration ensures 11 x 9's of data durability, and the self-healing technology can be configured to protect against silent data corruption or bit rot by performing an integrity check of all objects within a bucket at least once a year.
Q:	Does Lyve Cloud guarantee permanent deletion of data?
A:	To permanently delete data, client applications should use S3 API calls to delete all objects and buckets that the application creates.
Q:	Does Lyve Cloud support multipart uploads?

A:	Lyve Cloud supports multipart uploads where the object part or chunk can range from 5 MB to 5 GB, with a maximum number of 10,000 parts per an object. This enables the efficient upload of large files and recovery from transmission errors.
Q:	What level of uptime does Lyve Cloud support?
A:	Lyve Cloud architecture implements redundancy at the hardware and software stack. All critical software components are designed to tolerate multiple failures. Lyve Cloud makes all reasonable efforts to maintain monthly uptime of 99.9% with applicable service credits. In addition, customers can use the replicated bucket feature to ensure there are separate copies of the data in different data centers.
Q:	Are there a minimum and maximum sizes for stored objects?
A:	There is no minimum object size, and the maximum object size for a single PUT operation is 5 TB. Lyve Cloud recommends using multipart uploads for files larger than 100 MB.
Q:	What kind of data can I store in Lyve Cloud?
A:	Lyve Cloud allows you to store any data in any format as long as it complies with Lyve Cloud's Terms and Conditions.
Q:	What versions of TLS / SSL does Lyve Cloud support?
A:	Lyve Cloud is compatible with TLS 1.2.
Q:	Does Lyve Cloud support WORM/Write-Once-Read-Many for data immutability?
A:	Lyve Cloud supports S3 object lock, which includes WORM/data immutability. This can be enabled at the bucket level.
Q:	Does Lyve Cloud support management and data access auditing?
A:	After enabling the Audit Log feature, Lyve Cloud keeps an audit log of all user access to your Lyve Cloud console and all client application's S3 operations. Users can designate a bucket to receive these audit logs from Lyve Cloud.
Q:	Can I connect Lyve Cloud with third-party tools?
A:	Yes. You can connect to Lyve Cloud using either CyberDuck, Rclone, or any S3 browser. Using these tools, create folders in buckets or upload files. Consult your organization's policies and the EULA policies of the software before downloading 3rd party applications. To connect to Lyve Cloud with third-party tools, see Connecting S3 clients.

Q:	Which operating systems environments are supported by the Lyve Cloud console?
A:	Lyve Cloud console supports Windows, Linux, and Mac operating systems. Lyve Cloud provides high standards of data protection and security. To ensure customer data is not compromised, we may not support clients running operating systems or application versions that are not actively supported by the relevant ISV or an open-source community.
Q:	Does Lyve Cloud support strong consistency?
A:	The Lyve Cloud S3 platform provides strong read-after-write consistency. This means that all **GET**, **PUT**, and **LIST** operations in S3 are consistent, ensuring that what you write is exactly what you will read, and that the results of a **LIST** operation will be a precise representation of the contents in the bucket. This feature is particularly useful for applications such as data lakes.

Buckets

Does Lyve Cloud have any object naming limitations?
Do buckets have a maximum storage limit?
How do I manage my buckets using API?
How do I utilize multipart upload and identify which parts of data have successfully been uploaded?
When should I use ListMultipartUploads API?
Does Lyve Cloud support versioned buckets?
How can a customer confirm the encryption status of their objects?
Can I check bucket utilization using commands?
How can I create a bucket with object immutability (lock) enabled with a set duration using S3 API?
How many buckets can you create in Lyve Cloud?

Q:	Does Lyve Cloud have any object naming limitations?
A:	The object name can contain special characters such as @ # * $ % & ! ? , ; ' " | + = < > ^ ( ) { } [ ], as well as alphanumeric characters like 0-9, a-z, A-Z . However, using any of these characters may cause issues due to limiting factors of the S3 client SDK..
Q:	Do buckets have a maximum storage limit?
A:	No. There is no storage limit for data stored in a single bucket in Lyve Cloud.
Q:	How do I manage my buckets using API?
A:	You can manage your bucket through the S3 API as well as the Lyve Cloud console. We support the standard S# bucket level commands.

Q:	How do I utilize multipart upload and identify which parts of data have successfully been uploaded?
A:	Lyve Cloud supports multipart uploads of objects up to 5TBs. Using this method, large files are broken into smaller pieces for a more efficient upload. The pieces are then put back together at the end of the process.A multipart upload consists of three steps: 1. **Multipart upload initiation** : When a multipart upload is initiated, AWS S3 will return an upload ID, which is a unique identifier for the multipart upload. This upload ID is a required field for all upload parts, list parts, complete upload or stop upload commands. 2. **Parts upload**: In this step, you can specify the part numbers which you would like to upload. A part number uniquely identifies a part and its place in the uploading object. For each part upload, please save the part number and ETag value. These will be needed to complete step 3. 3. **Multipart upload completion** : In this step, S3 will complete the upload by piecing the parts in order based on the part number. After a successful complete request, the individual parts will no longer exist. If an object is above 5TB in size, then the multipart upload completion command will not succeed. If a multi-part upload fails, the upload can resume with the part of the upload which failed. To view which parts of an upload succeeded, use the ListParts command. This will return all uploaded parts with their size and each one's part number. For more information, see: ListParts .
Q:	When should I use ListMultipartUploads API?
A:	List multipart upload lists the in-progress multipart uploads that are initiated but are not yet completed or aborted. This API allows writing code that will the uploads that are not completed successfully on time. Lyve Cloud is performing this automatically, by cancelling all pending multi parts after 24 hours. If desired, you can control this setting via the use of bucket policy rules.
Q:	Does Lyve Cloud support versioned buckets?
A:	Yes, versioning is fully supported and can be enabled when creating a bucket.
Q:	How can a customer confirm the encryption status of their objects?

A:	Lyve Cloud enforces standard TLS 1.2 with 256-bit advanced encryption standard (AES) Galois/Counter Mode (GCM)—otherwise known as AES-256-GCM—to establish secure communications to the customer in transit and at rest. As an authenticated encryption algorithm, GCM provides proven security of the symmetric-key cryptographic cipher that has wide adoption for its performance. Seagate storage hardware is validated by Federal Information Processing Standards (FIPS) 140-2/3, which directly aligns with the Lyve Cloud focus on security and performance.To view the objects encryption status, please follow the steps below.
	
	Pre-requisites
	
	• Download a command-line tool such as AWS CLI. See using the AWS S3 command line.
	
	Open your command line application (Command Prompt for PC, Terminal for Mac) and use the following command.
	
	`--profile profile name --endpoint http://s3.<endpoint-URL> s3api head-object--bucket bucket name--key file name`
	
	`C:\Users\515515>aws --profile Kevin --endpoint https://s3.us-east-1.lyvecloud.seagate.com s3api head-object --bucket brawleytest --key Goals.docx`
	
	Result
	
	`bytes 20900 application/vnd.openxmlformats--officedocument.wordprocessingml.document "34162b3bec92a8334bd9fca388477f85" Mon, 06 Dec 2021 21:10:00 GMT AES256 bytes 20900 application/vnd.openxmlformats--officedocument.wordprocessingml.document "34162b3bec92a8334bd9fca388477f85" Mon, 06 Dec 2021 21:10:00 GMT AES256 METADATA 20210828T175628Z 1b6a154b13045741f4b61ab07ed55567754f44aa6796cc250b2a506c6c83a11a METADATA 20210828T175628Z 1b6a154b13045741f4b61ab07ed55567754f44aa6796cc250b2a506c6c83a11a`
	
	Note—The encryption is shown here as AES256 which is highlighted in bold.

Q:	Can I check bucket utilization using commands?

A:	The content and data quantity in a bucket can be viewed through the following AWS CLI commands.
	
	`--profile profile name –endpoint URL s3 ls –summarize –human-readable –recursive s3://bucket`
	
	`C:\Users\515515>aws –profile sv15 –endpoint https://s3.us-west-1.lyvecloud.seagate.com s3 ls –summarize –human-readable –recursive s3://mybuck`
	
	Result
	
	2021-07-03 22:06:34 6 Bytes my-test-file.txt
	2021-07-03 22:07:48 12 Bytes my-test-file1.txt
	2021-07-03 22:29:33 11 Bytes my-test-file2.txt
	2021-07-01 00:46:18 531 Bytes service-acounts.txt
	Total Objects: 4 Total Size: 560 Bytes
	
	Note—This command will list all contents in the bucket.

Q:	How can I create a bucket with object immutability (lock) enabled with a set duration using S3 API?

A:	1. Create a bucket and enable object immutability using the following command:
	
	`aws s3api create-bucket --bucket <bucket name> --object-lock-enabled-for-bucket --profile <profile name> --endpoint <endpoint>`
	
	`aws --profile va3 --endpoint https://s3.us-east-1.lyvecloud.seagate.com s3api create-bucket --bucket mybucket --object-lock-enabled-for-bucket`
	
	Note—You can enable object immutability only while creating a bucket.
	
	Response
	
	`{ "Location": http://s3.us-east-1.lyvecloud.seagate.com/my-bucket-with-object-lock }`
	
	2. Set the duration using the following command:
	
	`aws s3api --profile <profile name> put-object-lock-configuration --bucket <bucketname> --object-lock-configuration <value> --endpoint <endpoint>`
	
	`aws --profile va3 --endpoint https://s3.us-east-1.lyvecloud.seagate.com s3api create-bucket --bucket mybucket --object-lock-enabled-for-bucket`
	
	Response
	
	`aws s3api --profile lcprd put-object-lock-configuration --bucket mybucket --object-lock-configuration ObjectLockEnabled="Enabled",Rule={DefaultRetention={Mode="COMPLIANCE",Days=1}} --endpoint https://s3.us-east-1.lyvecloud.seagate.com`

Q:	How many buckets can you create in Lyve Cloud?
A:	You can create an unlimited number of buckets per account.

Service Account

What if I lose my S3 key credentials?

Q:	What if I lose my S3 key credentials?
A:	As a reminder, your S3 key allows applications to authenticate and access Lyve Cloud buckets and objects. The access key and secret key (credentials) are generated when you request a key. This information must be saved at the time of account creation and cannot be recovered afterwards. If you lose or misplace your S3 key and believe you need new credentials, you must create a new key. You may have multiple (unlimited) keys for a specific account or user. You may also delete old keys as needed. Note—If you delete a key, any application using that will not have access to the data until you configure a valid key.

User and Roles

Can I switch admin roles in Lyve Cloud?
Can I change the root user of an account?
What if I forget my password for the console?

Q:	Can I switch admin roles in Lyve Cloud?
A:	Yes. If you are an administrator-level user and would like to change your or another user's permissions, you can set them in the console.
Q:	Can I change the root user of an account?
A:	Not through the standard user interface. You need to open a ticket with Lyve support (lyve.support@seagate.com).
Q:	What if I forget my password for the console?
A:	If your user account is an email address, then you can select the Forgot password link and have the system email you a link to reset your account password. If your account is not an email address, contact Lyve support (support.lyve@seagate.com) to assist you.

Self Service

How do I sign up for Lyve Cloud?
I cannot register for Lyve Cloud services using a credit card and get the following screen during registration.
I cannot access the registration link and receive the error The Site Can't be Reached.

After registration, the Lyve Cloud site fails to load with a Registration failed message.
When accessing the registration link, an unexpected error occurred.
How do I update or edit my credit card information?
When can I see my invoice for this month?
I did not receive the email with the confirmation code. How do I proceed?
I did not receive the email with a confirmation of account creation. How do I proceed?
How do I cancel my Lyve Cloud account?
When will my Credit Card be charged after the cancellation?
Why does my credit card provider show a transaction as "pending" from Lyve Cloud on my customer bill after I have registered for the free trial?

Q:	How do I sign up for Lyve Cloud?
A:	You can sign up for Lyve Cloud using https://lyve.seagate.com/.
Q:	I cannot register for Lyve Cloud services using a credit card and get the following screen during registration. [INSERT faq-cannot-register-with-credit-card-01.png]
A:	The Lyve Cloud subscription has reached the limit. You can wait for a period of time and try again. If you are still facing issues, please contact our support team at lyve.support@seagate.com.
Q:	I cannot access the registration link and receive the error **The Site Can't be Reached**.
A:	You can perform the following actions and retry: • Check that your internet connection is working or restarting your router or modem often resolves this problem. • Disable your firewall and antivirus software. • Clear your browser cache. • Check your DNS server settings.
Q:	After registration, the Lyve Cloud site fails to load with a Registration failed message. [INSERT faq-registration-failed-01.jpg]
A:	The registration could have failed because the credit card authorization was incomplete or the credit card was declined. If the authorization is successful and still the registration fails, please contact our support team at lyve.support@seagate.com.
Q:	When accessing the registration link, an unexpected error occurred. [INSERT faq-unexpected-error-01.jpg]

A:	Please contact our support team at lyve.support@seagate.com.
Q:	How do I update or edit my credit card information?
A:	To update or edit credit card information such as the expiration date, see TBD.
Q:	When can I see my invoice for this month?
A:	Your invoice is generated on the first of every month for the previous month.
Q:	I did not receive the email with the confirmation code. How do I proceed?
A:	Please check your inbox for an email entitled **Lyve Cloud verification code**. Be sure to also check your spam folder. If you still haven't received the email, please contact our support team lyve.support@seagate.com.
Q:	I did not receive the email with a confirmation of account creation. How do I proceed?
A:	Please check your inbox for an email entitled **You are invited to Lyve Cloud!**. Be sure to also check your spam folder. If you still haven't received the email, please contact our support team lyve.support@seagate.com.
Q:	How do I cancel my Lyve Cloud account?
A:	Lyve Cloud customers can close their own accounts.
Q:	When will my Credit Card be charged after the cancellation?
A:	When your account is cancelled, the billing period ends on the cancellation date. The bill is generated, and the credit card is charged on the cancellation date at the end of the day.
Q:	Why does my credit card provider show a transaction as "pending" from Lyve Cloud on my customer bill after I have registered for the free trial?
A:	Lyve Cloud only authorizes charges during registration and does not execute an actual charge to your credit card. Some credit card companies elect to note this authorization on customer statements as a **pending** charge. Not all companies follow this practice. While you may see a charge from Lyve Cloud as pending, you will not be charged for Lyve Cloud service during the free trial period. Please speak with your credit card company about concerns with this notation on your customer bill. You may also contact Lyve Cloud support for further assistance.

Billing

What storage classes does Lyve Cloud offer?

How much does it cost to use Lyve Cloud?
How is the capacity utilization metric calculated?
What is the cost impact of using Lyve Cloud's immutability feature?
What is the cost impact of using Lyve Cloud's audit logging features?
I'm not planning to use the object immutability feature but I may be overwriting a file with the same name multiple times. What is the cost impact of this?
Are there any charges for egress or API requests in Lyve Cloud?
Can I get an extension of a free trial?
I am currently using the free trail, how can I become a paid customer?
How is Lyve Cloud billing information collected, and how do I get billed?
Where can I get a copy of my invoice?

Q:	What storage classes does Lyve Cloud offer?
A:	Lyve Cloud offers a single storage tier and a single consistent price based on the amount of data stored.
Q:	How much does it cost to use Lyve Cloud?
A:	The Lyve Cloud service is based on a simple pricing model. Lyve Cloud charges a monthly fee based on the average capacity of data stored during a given month. You are not charged for S3 API calls or data egress.
Q:	How is the capacity utilization metric calculated?
A:	Lyve Cloud measures the total storage consumption four times a day. Storage consumption number recorded is the average of the four instances per day. The monthly average usage is calculated by taking the average of all daily records for a given month. For example, if the daily average for the first 10 days of the month is 10 TB, the average for next 10 days is 20 TB and the average for last 10 days of the month is 30 TB, then the total consumption for the month is 20 TB ((10+20+30)/3=20).
Q:	What is the cost impact of using Lyve Cloud's immutability feature?
A:	Once object immutability is switched on, bucket versioning is automatically enabled. Updating a file creates a new version of the objects being stored. This process results in an increase in storage usage and cost.
Q:	What is the cost impact of using Lyve Cloud's audit logging features?
A:	Enabling audit logging creates log files for buckets or console activity. These log files are stored and treated like all other billable storage.
Q:	I'm not planning to use the object immutability feature but I may be overwriting a file with the same name multiple times. What is the cost impact of this?

A:	Let us consider an example to answer this question.On day 1, you store a file called "file1.txt". On day 2, you then overwrite "file1.txt" with a new copy of "file1.txt" while not changing the file name. The original copy of "file1.txt" is, therefore, overwritten and the cost is impacted only by changes in the file size of the latest “file1.txt” that may have occurred.
Q:	Are there any charges for egress or API requests in Lyve Cloud?
A:	There are no egress or API charges in Lyve Cloud.
Q:	Can I get an extension of a free trial?
A:	Yes, please use the extend option on your trial banner or contact your Lyve Cloud sales representative for an extension.
Q:	I am currently using the free trail, how can I become a paid customer?
A:	Please contact your Lyve Cloud sales representative to become a paid customer.
Q:	How is Lyve Cloud billing information collected, and how do I get billed?
A:	Invoices reflect a calculation of the average monthly storage usage for the previous month. Invoices are available on the first day of every month. You then have an agreed time period to complete payment.
Q:	Where can I get a copy of my invoice?
A:	You can view and download a copy of your invoice from the Lyve Cloud console.

Security

How is my data protected?
Does Lyve Cloud mine my data?
What happens to my data if I am no longer a Lyve Cloud customer?
What authentication mechanisms are supported?
How secure are Lyve Cloud datacenters?
Which OTP applications can be used for MFA login?
Can I use CORS with Lyve Cloud?
How can I change my authentication method?
Why must I use a mobile phone to set up MFA?
My mobile device with authenticator app is lost or stolen, what do I do?
Can Email be used as the 2nd method of Auth for MFA?
Can organizations use their own SAML with MFA?
I have lost my recovery code, how do I login to the Lyve Cloud console?
How can I change my Multi-Factor Authentication (MFA) Phone Number?

Can I change my authenticator app?
Can I register multiple Lyve Cloud accounts in an authenticator app for MFA?

Q:	How is my data protected?
A:	Lyve Cloud protects customer data in transit by using: • Transport layer security (TLS) for data in-flight TLS 1.2 (AES-256-GCM) • Encryption for data at rest The data is always encrypted at rest using one of two server-side methods: • Encryption with a client-provided key (part of S3 request headers) - SSE-C • Encryption with an S3 managed encryption keys - SSE-S3 All data, regardless of whether it is encrypted or not on the client-side (SSE-C), is encrypted using AES 256-bit encryption at rest. The keys are never shared and can be rotated based on the customer's security policy. Data in flight is encrypted using TLS 1.2, and client applications can only connect using HTTPS protocol.Lyve Cloud follows industry best practices for design and security models. Contact sales.lyvecloud@seagate.com for a complete overview of security analysis conducted by a third party.
Q:	Does Lyve Cloud mine my data?
A:	No, Lyve Cloud does not mine any customer data. All data stored in Lyve Cloud is encrypted. We strongly recommend that customers use client-side encryption for complete data protection.
Q:	What happens to my data if I am no longer a Lyve Cloud customer?
A:	Lyve Cloud does not make any secondary copies of the data. To permanently delete the data, the client application should use the S3 API calls to delete all objects and buckets it created. Once this is complete, customers can emailsupport.lyvecloud@seagate.com to request that their account information be permanently deleted. This ensures that any remaining customer information is removed from the Lyve Cloud cluster.
Q:	What authentication mechanisms are supported?
A:	Access to the Lyve Cloud admin portal is supported by multiple authentication schemes, including: • Multi-factor authentication using either SMS OTP (One Time Password) or an authenticator mobile app • Federated login using the customer's IDP login flow

Q:	How secure are Lyve Cloud datacenters?
A:	Lyve Cloud prioritizes a secure and protected infrastructure. • A dedicated staff manages and protects each site 24×7, year-round. • Each site is equipped with security cameras to monitor inside the data center and the surrounding area. • Facilities are unmarked so as to not draw attention from outside. • Building access is controlled using biometric measures.
Q:	Which OTP applications can be used for MFA login?
A:	Lyve Cloud supports the use of third-party authenticator apps as verification methods for MFA logins. You can use any authenticator app that generates temporary codes based on the time-based one-time password. There are many free and paid authenticator apps to choose from. Widely-used options include Google Authenticator, Microsoft Authenticator, DUO, Authy, Okta Verify, Auth0 Guardian, OneLogin Protect, and Oracle Authenticator.
Q:	Can I use CORS with Lyve Cloud?
A:	Yes, Cross-Origin Resource Sharing (CORS) is fully supported.
Q:	How can I change my authentication method?
A:	You must contact your administrator to reset MFA for the user. After resetting MFA, you must again enroll in MFA. For more information, see Enrolling in MFA.
Q:	Why must I use a mobile phone to set up MFA?
A:	Your device is unique to you. This helps to ensure that your account can only be accessed by the person in possession of your phone. Even if someone has your Lyve Cloud credentials, they will not be able to access your Lyve Cloud account without your mobile phone.
Q:	My mobile device with authenticator app is lost or stolen, what do I do?
A:	To change your phone number you must contact the administrator of your account to reset MFA. After the administrator resets MFA, you must again enroll in MFA on your new device. For more information, see Enrolling in MFA.
Q:	Can Email be used as the 2nd method of Auth for MFA?

A:	No, email is not supported as an MFA method. We only support authenticator apps because email credentials can be easily compromised or reset. With a mobile device, an authenticator app adds another layer of security when accessing your account.
Q:	Can organizations use their own SAML with MFA?
A:	Yes, organizations can use their own SAML with MFA. Lyve Cloud MFA always applies to password users even if federated login is enabled for the account.
Q:	I have lost my recovery code, how do I login to the Lyve Cloud console?
A:	You must contact your administrator to reset MFA. After resetting MFA, you must again enroll in MFA. For more information, seeEnrolling in MFA.
Q:	How can I change my Multi-Factor Authentication (MFA) Phone Number?
A:	An administrator must reset the MFA for a user to change the associated phone number. After resetting MFA, you must again enroll for MFA. For more information, seeEnrolling in MFA.
Q:	Can I change my authenticator app?
A:	Yes, you can change the authenticator app by installing the preferred authenticator app.You must contact your administrator to reset MFA for the user. After resetting MFA, you must again enroll in MFA. For more information, seeEnrolling in MFA.
Q:	Can I register multiple Lyve Cloud accounts in an authenticator app for MFA?
A:	You may use different authenticator apps for different Lyve Cloud accounts. If you are required to use the same authenticator app for multiple Lyve Cloud accounts, refer to the MFA application's help section to learn how to add multiple accounts. Follow the steps based on the desired authenticator app.Refer to few of the commonly used authenticator apps: • Google • Microsoft • Oracle

Support

How can I contact Support?
What can I expect after a support ticket is raised or an email is sent?
What details are required when contacting support through email?
When can I contact Lyve Cloud Support?

When will the support team contact me after I create a support ticket?
Where to find help?

Q:	How can I contact Support?
A:	There are two ways to contact Lyve Cloud Support. 1. Create a support ticket from the Lyve Cloud console. 2. Email lyve.support@seagate.com
Q:	What can I expect after a support ticket is raised or an email is sent?
A:	**Communication** Once you submit a ticket, you will get an auto-generated email from lyve.support@seagate.com that includes your ticket number. Your ticket is added to a queue and is processed in the order that it is received. However, support times do vary depending on issue severity. If you need to contact support for any reason regarding your ticket, you can reply to the auto-generated email, or you can leave a comment in the portal under your ticket details. **SLA** We will use reasonable efforts to provide a workaround within 24 hours. Afterwards, we will do a root cause analysis and fix the issue to ensure it does not occur again. **Escalation** Lyve Cloud uses an internal escalation process to ensure tickets are resolved quickly and that they receive the right technical expertise. Following this process, there may be cases where customers learn their ticket has been escalated to another part of the support organization. **Resolution** When the ticket is resolved, our support team will send out a post-support survey to assess the effectiveness of our support program. Your feedback is highly valuable to us.
Q:	What details are required when contacting support through email?

A:	When emailing support, you must include as much information as possible. Please include the following in your request: • Your account ID and endpoint. It can be found on your Lyve Cloud URL - <account_id>.console.lyvecloud.seagate.com • Summary and description of the issue you are experiencing • Environment (for example, us-east-1 (Virginia), us-west-1 (California), ap-southeast-1 (Singapore), and so on) • URL link • Attachment with an error screen, if any
Q:	When can I contact Lyve Cloud Support?
A:	Lyve Cloud support teams are staffed around the globe, working 24/7, 365 days a year.
Q:	When will the support team contact me after I create a support ticket?
A:	A new ticket will trigger an auto-response acknowledgement that you will receive within 30 minutes. Our support team will revert you within 4 hours. Seagate will use reasonable efforts to provide a workaround within 24 hours. Later, we will do a root cause analysis and fix the issue to ensure it does not occur again.
Q:	Where to find help?
A:	Before contacting support, it may be quick and convenient to find the information you need in our documentation. The Lyve Cloud Documentation provides several resources to answer your questions. Resources include our Quick Start Guide, Video tutorials, Connecting to S3 clients, Known Issues, FAQs, and much more.

Partner

Where can I view my customers' storage activity?
Where can I view my bill?
How can I calculate storage cost per customer?

Q:	Where can I view my customers' storage activity?
A:	Lyve Cloud partners can view the monthly storage activity of their customers by viewing the Usage Trend dashboard on the Lyve Cloud console.Total number of users, buckets, service accounts, average usage, and support tickets can also be viewed on the Customer distributions dashboard on the Lyve Cloud console.
Q:	Where can I view my bill?

A:	Your bill is sent to the email address registered with the partner account.
Q:	How can I calculate storage cost per customer?
A:	Invoices feature monthly storage used by customers to let partners determine to price for each customer. Lyve Cloud Partner Portal provides usage by customers that can be downloaded from the Home page dashboard. This enables partners to determine to price for each of their customers.

HIPAA

What is HIPAA?
Is Seagate Lyve Cloud HIPAA compliant?
What measures does Seagate use to ensure its service is HIPAA compliant?
Where can I learn more about HIPAA?

Q:	What is HIPAA?
A:	The Health Insurance Portability and Accountability Act (HIPAA) is a US law enacted in 1996. HIPAA is a comprehensive set of standards that regulate the protection and use of protected health information (PHI) in the healthcare industry. The law applies to healthcare providers, health plans, and healthcare clearinghouses that transmit health information in electronic form. HIPAA establishes national standards for the privacy and security of PHI and requires that these entities implement appropriate administrative, physical, and technical safeguards to protect the confidentiality, integrity, and availability of PHI. HIPAA also sets rules for the use and disclosure of PHI and gives individuals certain rights with respect to their PHI, such as the right to access, inspect, and receive a copy of their PHI. In summary, HIPAA is a comprehensive set of standards and rules that regulate the protection and use of PHI in the healthcare industry and establishes national standards for the privacy and security of PHI.
Q:	Is Seagate Lyve Cloud HIPAA compliant?
A:	Yes, Seagate's Lyve Cloud storage service has been audited by trusted third-party certifiers and is certified to comply with all HIPAA standards for the secure handling and protection of PHI.
Q:	What measures does Seagate use to ensure its service is HIPAA compliant?

A:	Seagate takes numerous measures to ensure that our service is compliant with HIPAA standards and provides a highly secure and robust data storage service. Such measures include:
	
	• Lyve Cloud is certified to the ISO/IEC 27001 Standard, as well as attested to the SOC 2 Type 2 Standard. We are also HIPAA compliant.
	
	  • ISO 27001 is the world's leading information security standard, providing control requirements to create an Information Security Management System (ISMS).
	  • SOC 2 is an extremely popular form of cybersecurity audit, used by a rapidly growing number of organizations to demonstrate they take cybersecurity and privacy seriously.
	  • HIPAA Security Rule addresses the safeguarding of ePHI through the application of administrative, physical, and technical safeguards. Compliance is required by all covered organizations defined by HIPAA and the Office of Civil Rights (OCR) audit guidelines and assessment standards.
	
	• Two-factor authentication for users to ensure secure data custodians.
	• Standardized TLS 1.2 data encryption.
	• Authentication and authorization within every data transaction, using both account access keys and a cryptographic signature.
	• Secure data deletion with Secure Erase. When a customer ends their service with Lyve Cloud, their data is securely cryptographically erased as the SSE-C key is only available to the customer in the API.
	• Enacting proper staff security awareness training and regulation policies, including procedures for authorizing access to PHI, as well as security incident response.
	• Entering into and documenting appropriate business associate contracts with covered entities using Lyve Cloud service.
	• Periodic, thorough risk analyses of current business practices/audits, with an aim to identify potential security risks and test current security and contingency policies.
	• Safe logging and monitoring procedures ensure the recording of all login attempts, individuals who access electronic PHI, as well as any security incidents and mitigation responses involved.
	• Extensive resilience testing and contingency backup architecture that prevents data loss in the case of a datacenter going down.

Q:	Where can I learn more about HIPAA?
A:	For more information regarding HIPAA compliance with Seagate's Lyve Cloud storage service, see Lyve Cloud HIPAA Business Associate Addendum

Lyve Cloud Compliance

This chapter provides a summary of the key certifications and compliance requirements that organizations should consider when selecting and implementing cloud services. From HIPAA to ISO 27001 to SOC 2, this guide covers the most widely recognized standards and best practices for cloud security and privacy. It explains why these certifications and requirements are critical to ensure the protection of sensitive information in the cloud.

HIPAA

Seagate designed Lyve Cloud to be a leading cloud storage solution for the healthcare sector, built on core principles of resilience, compliance, performance, and value. Recognized by Health and Human Services (HHS.org) as a No View SaaS provider, Lyve Cloud ensures that customer data remains fully protected. Lyve Cloud's standards-based architecture enables exceptional levels of security and regulatory compliance, setting it apart in the cloud storage market.

Lyve Cloud delivers high-availability, hot-tier object storage at the cost of archival storage, offering unmatched value for healthcare organizations. Unlike legacy providers, Lyve Cloud was built with modern security-first principles, resulting in fewer vulnerabilities and a more robust infrastructure.

Lyve Cloud has a HIPAA Compliant report

A HIPAA (Health Insurance Portability and Accountability Act) compliant cloud is expected to meet certain standards to ensure the protection of sensitive health information. The following are some of the key expectations of a HIPAA-compliant cloud:

- **Security**: The provider must have strong security measures in place to protect the confidentiality, integrity, and availability of electronically protected health information (ePHI). This includes encryption, access controls, and audit logs.
- **Privacy**: The provider must have strict privacy policies and procedures in place to ensure that ePHI is only accessed by authorized individuals. This includes limiting access to ePHI, conducting background checks on employees, and training employees on privacy and security.
- **Compliance**: The provider must comply with all HIPAA regulations, including the HIPAA Security Rule, the HIPAA Breach Notification Rule, and associated policies, procedures, and documentation.
- **Business Associate Agreement**: The provider must sign a Business Associate Agreement (BAA) with its clients to ensure that they understand and agree to comply with HIPAA regulations.
- **Disaster Recovery and Business Continuity**: The provider must have a disaster recovery and business continuity plan in place to ensure that ePHI can be recovered in the event of a disaster or data loss.
- **Monitoring and Auditing**: The provider must regularly monitor and audit systems and processes to ensure that they comply with HIPAA regulations and to identify and address any potential security or privacy breaches.
- **Technical Support**: The provider must provide technical support to their clients to ensure that they

can effectively use the cloud and resolve any issues that may arise.

It is important to note that the responsibility for ensuring HIPAA compliance does not rest solely with the cloud provider. The entity using the cloud (known as the covered entity) must also ensure that they comply with HIPAA regulations.

ISO 27001

ISO 27001 is an internationally recognized standard for information security management. It defines a framework of best practices and controls designed to safeguard sensitive data. A cloud service certified under ISO 27001 is expected to meet the following core requirements:

- **Information Security Management System (ISMS)**: The cloud provider must maintain a comprehensive ISMS that governs the protection of sensitive information. This system should encompass all facets of security, including access control, incident response, risk management, and business continuity.
- **Security Controls**: Robust security controls must be in place to defend against threats. These include, but are not limited to, encryption, access management, firewalls, and intrusion detection/prevention systems.
- **Risk Management**: A proactive risk management process is essential. The provider must regularly identify, assess, and mitigate risks through security assessments, threat modeling, and the implementation of appropriate safeguards.
- **Data Privacy**: Strict data privacy policies and procedures must ensure that sensitive information is accessed only by authorized personnel and handled in compliance with relevant regulations.
- **Business Continuity and Disaster Recovery**: The cloud service must have tested plans for business continuity and disaster recovery to ensure data protection and service availability during disruptions or data loss events.
- **Monitoring and Auditing**: Continuous monitoring and regular audits are required to verify compliance with ISO 27001 and to detect and respond to potential security or privacy incidents.
- **Technical Support**: The cloud service must provide technical support to their clients to ensure that they can effectively use the cloud and resolve any issues that may arise.
- **Continual Improvement**: The provider must implement a continuous improvement process to ensure its security controls and procedures are regularly updated to address emerging threats and evolving risks.

A cloud service with an ISO 27001 certificate is expected to have a comprehensive and robust approach to information security management covering all aspects of information security, including but not limited to risk management, data privacy, business continuity, and monitoring and auditing.

Type 2 SOC 2

Lyve Cloud has a Type 2 SOC 2 Attestation report

A SOC 2 attestation is a third-party assessment of a cloud service provider's controls related to the

security, availability, processing integrity, confidentiality, and privacy of the information processed by the service. Type 2 SOC 2 attestation specifically refers to an assessment of the cloud service provider's controls over a period of time (typically six months or more).

A cloud service provider that has received a Type 2 SOC 2 attestation is expected to meet the following expectations:

- **Information Security**: A robust security program must be in place to safeguard sensitive data. This includes implementing access controls, encryption, firewalls, and regular monitoring and auditing of systems.
- **Availability**: The provider must ensure high service availability through redundant infrastructure, real-time monitoring, and a well-defined disaster recovery strategy.
- **Processing Integrity**: Controls must be implemented to guarantee the accuracy and reliability of data processing. This includes validation mechanisms, error detection, and audit trails.
- **Confidentiality**: Strict policies and procedures must protect sensitive information, ensuring access is limited to authorized individuals only.
- **Privacy**: A comprehensive privacy program must be in place to meet the requirements of relevant privacy regulations and standards such as GDPR or HIPAA.
- **Monitoring and Auditing**: The provider must regularly monitor and audit systems and processes to ensure that they comply with SOC 2, and to identify and address potential security or privacy breaches.
- **Technical Support**: Clients must have access to technical support to ensure that they can effectively use their services and resolve issues.
- **Continual Improvement**: The provider must maintain a continuous improvement process to adapt its security controls and practices to evolving threats and industry standards.

Lyve Cloud's Type 2 SOC 2 attestation reflects a comprehensive and proactive approach to information security, privacy, and availability. It assures customers that their data is protected by industry-leading controls and that the platform is built for trust, resilience, and compliance.

Summary

Certifications and compliance are essential to establishing trust and confidence in Lyve Cloud. They demonstrate alignment with legal and regulatory standards, reinforce strong security practices, support effective risk management, and offer a competitive edge in the cloud services market.

Lyve Cloud holds several key international certifications and attestations, reflecting its commitment to security, privacy, and operational excellence. This list continues to grow, driven by ongoing customer feedback and evolving industry requirements.