Use a tenant account

StorageGRID software

NetApp
December 03, 2025

Table of Contents

Use a tenant account

Use a tenant account

A tenant account allows you to use either the Simple Storage Service (S3) REST API or the Swift REST API to store and retrieve objects in a StorageGRID system.

What is a tenant account?

Each tenant account has its own federated or local groups, users, S3 buckets or Swift containers, and objects.

Tenant accounts can be used to segregate stored objects by different entities. For example, multiple tenant accounts can be used for either of these use cases:

- **Enterprise use case:** If the StorageGRID system is being used within an enterprise, the grid's object storage might be segregated by the different departments in the organization. For example, there might be tenant accounts for the Marketing department, the Customer Support department, the Human Resources department, and so on.

  > If you use the S3 client protocol, you can also use S3 buckets and bucket policies to segregate objects between the departments in an enterprise. You don't need to create separate tenant accounts. See instructions for implementing S3 buckets and bucket policies for more information.

- **Service provider use case:** If the StorageGRID system is being used by a service provider, the grid's object storage might be segregated by the different entities that lease the storage. For example, there might be tenant accounts for Company A, Company B, Company C, and so on.

How to create a tenant account

Tenant accounts are created by a StorageGRID grid administrator using the Grid Manager. When creating a tenant account, the grid administrator specifies the following:

- Basic information including the tenant name, client type (S3) and optional storage quota.
- Permissions for the tenant account, such as whether the tenant account can use S3 platform services, configure its own identity source, use S3 Select, or use a grid federation connection.
- The initial root access for the tenant, based on whether the StorageGRID system uses local groups and users, identity federation, or single sign-on (SSO).

In addition, grid administrators can enable the S3 Object Lock setting for the StorageGRID system if S3 tenant accounts need to comply with regulatory requirements. When S3 Object Lock is enabled, all S3 tenant accounts can create and manage compliant buckets.

Configure S3 tenants

After an S3 tenant account is created, you can access the Tenant Manager to perform tasks such as the following:

- Set up identity federation (unless the identity source is shared with the grid)
- Manage groups and users

- Use grid federation for account clone and cross-grid replication
- Manage S3 access keys
- Create and manage S3 buckets
- Use S3 platform services
- Use S3 Select
- Monitor storage usage

Although you can create and manage S3 buckets with the Tenant Manager, you must use an S3 client or S3 Console to ingest and manage objects.

How to sign in and sign out

Sign in to Tenant Manager

You access the Tenant Manager by entering the URL for the tenant into the address bar of a supported web browser.

Before you begin

- You have your login credentials.
- You have a URL for accessing the Tenant Manager, as supplied by your grid administrator. The URL will look like one of these examples:

  ```
  https://FQDN_or_Admin_Node_IP/

  https://FQDN_or_Admin_Node_IP:port/

  https://FQDN_or_Admin_Node_IP/?accountId=20-digit-account-id

  https://FQDN_or_Admin_Node_IP:port/?accountId=20-digit-account-id
  ```

  The URL always includes a fully qualified domain name (FQDN), the IP address of an Admin Node, or the virtual IP address of an HA group of Admin Nodes. It might also include a port number, the 20-digit tenant account ID, or both.

- If the URL does not include the tenant's 20-digit account ID, you have this account ID.
- You are using a supported web browser.
- Cookies are enabled in your web browser.
- You belong to a user group that has specific access permissions.

Steps

1. Launch a supported web browser.
2. In the browser's address bar, enter the URL for accessing Tenant Manager.
3. If you are prompted with a security alert, install the certificate using the browser's installation wizard.
4. Sign in to the Tenant Manager.

   The sign-in screen that appears depends on the URL you entered and whether single sign-on (SSO) has

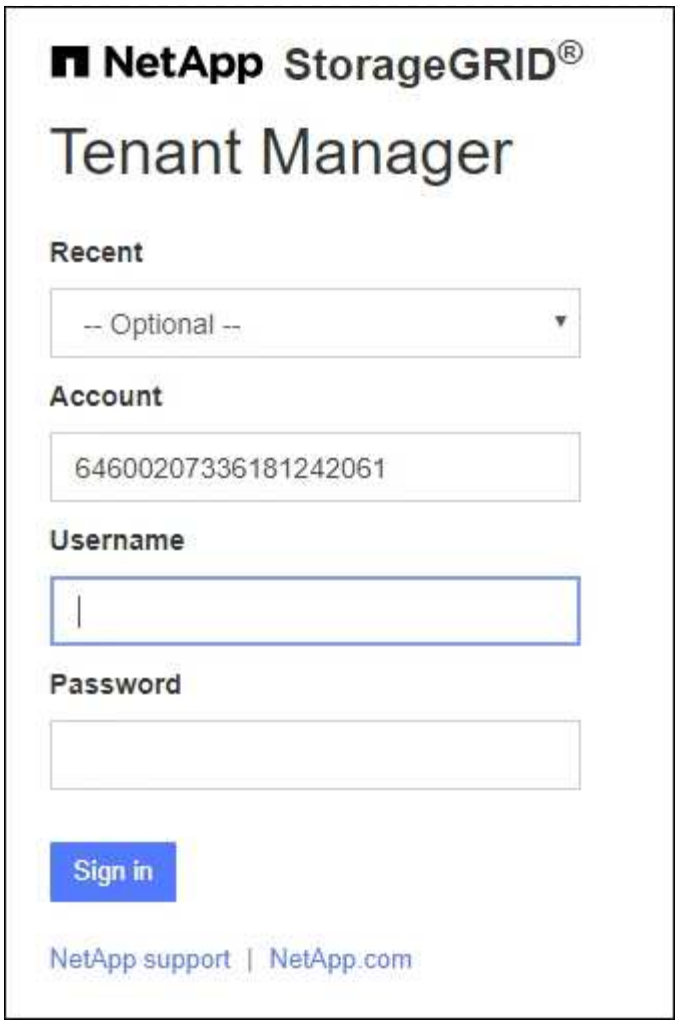
been configured for StorageGRID.

Not using SSO

If StorageGRID is not using SSO, one of the following screens appears:

- The Grid Manager sign-in page. Select the **Tenant sign-in** link.

- The Tenant Manager sign-in page. The **Account** field might already be completed, as shown below.



a. If the tenant's 20-digit account ID is not shown, select the name of the tenant account if it appears in the list of recent accounts, or enter the account ID.
b. Enter your username and password.
c. Select **Sign in**.

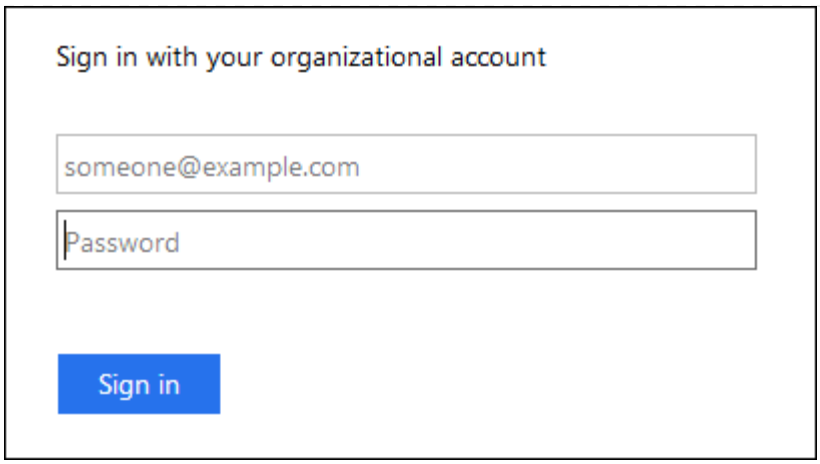
   The Tenant Manager dashboard appears.

d. If you received an initial password from someone else, select ***username*** > **Change password** to secure your account.

Using SSO

If StorageGRID is using SSO, one of the following screens appears:

- Your organization's SSO page. For example:



Enter your standard SSO credentials, and select **Sign in**.

- The Tenant Manager SSO sign-in page.

a. If the tenant's 20-digit account ID is not shown, select the name of the tenant account if it appears in the list of recent accounts, or enter the account ID.
b. Select **Sign in**.
c. Sign in with your standard SSO credentials on your organization's SSO sign-in page.

   The Tenant Manager dashboard appears.

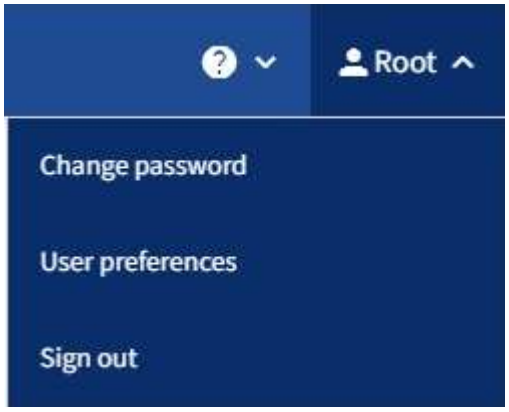
Sign out of Tenant Manager

When you are done working with the Tenant Manager, you must sign out to ensure that unauthorized users can't access the StorageGRID system. Closing your browser might

not sign you out of the system, based on browser cookie settings.

Steps

1. Locate the username drop-down in the top-right corner of the user interface.



2. Select the username and then select **Sign out**.
   - If SSO is not in use:

     You are signed out of the Admin Node. The Tenant Manager sign in page is displayed.

     If you signed into more than one Admin Node, you must sign out of each node.

   - If SSO is enabled:

     You are signed out of all Admin Nodes you were accessing. The StorageGRID Sign in page is displayed. The name of the tenant account you just accessed is listed as the default in the **Recent Accounts** drop-down, and the tenant's **Account ID** is shown.

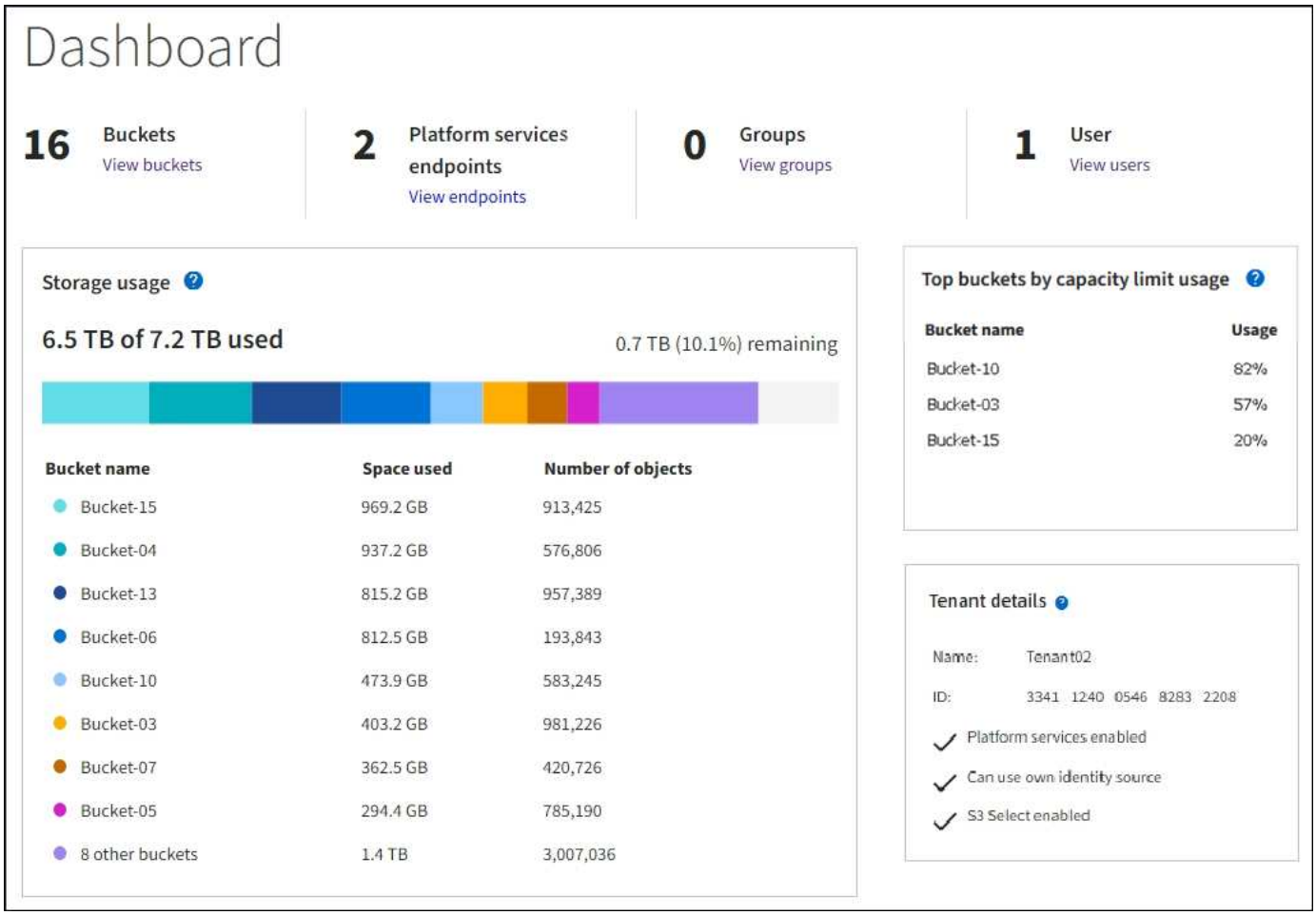
     If SSO is enabled and you are also signed in to the Grid Manager, you must also sign out of the Grid Manager to sign out of SSO.

Understand Tenant Manager dashboard

The Tenant Manager dashboard provides an overview of a tenant account's configuration and the amount of space used by objects in the tenant's buckets (S3) or containers (Swift). If the tenant has a quota, the dashboard shows how much of the quota is used and how much is remaining. If there are any errors related to the tenant account, the errors are shown on the dashboard.

The Space used values are estimates. These estimates are affected by the timing of ingests, network connectivity, and node status.

When objects have been uploaded, the dashboard looks like the following example:



Tenant account information

The top of the dashboard displays the number of configured buckets or containers, groups, and users. It also displays the number of platform services endpoints, if any have been configured. Select the links to view the details.

Depending on the tenant management permissions you have and the options you've configured, the remainder of the dashboard displays various combinations of guidelines, storage usage, object information, and tenant details.

Storage and quota usage

The Storage usage panel contains the following information:

- The amount of object data for the tenant.

  This value indicates the total amount of object data uploaded and does not represent the space used to store copies of those objects and their metadata.

- If a quota is set, the total amount of space available for object data and the amount and percentage of space remaining. The quota limits the amount of object data that can be ingested.

Quota usage is based on internal estimates and might be exceeded in some cases. For example, StorageGRID checks the quota when a tenant starts uploading objects and rejects new ingests if the tenant has exceeded the quota. However, StorageGRID does not take into account the size of the current upload when determining if the quota has been exceeded. If objects are deleted, a tenant might be temporarily prevented from uploading new objects until the quota usage is recalculated. Quota usage calculations can take 10 minutes or longer.

- A bar chart that represents the relative sizes of the largest buckets or containers.

  You can place your cursor over any of the chart segments to view the total space consumed by that bucket or container.

- To correspond with the bar chart, a list of the largest buckets or containers, including the total amount of object data and the number of objects for each bucket or container.

Bucket name	Space used	Number of objects
Bucket-02	944.7 GB	7,575
Bucket-09	899.6 GB	589,677
Bucket-15	889.6 GB	623,542
Bucket-06	846.4 GB	648,619
Bucket-07	730.8 GB	808,655
Bucket-04	700.8 GB	420,493
Bucket-11	663.5 GB	993,729
Bucket-03	656.9 GB	379,329
9 other buckets	2.3 TB	5,171,588

  If the tenant has more than nine buckets or containers, all other buckets or containers are combined into a single entry at the bottom of the list.

To change units for the storage values displayed in the Tenant Manager, select the user drop-down in the upper right of the Tenant Manager, then select **User preferences**.

Quota usage alerts

If quota usage alerts have been enabled in the Grid Manager, these alerts will appear in the Tenant Manager when the quota is low or exceeded, as follows:

- If 90% or more of a tenant's quota has been used, the **Tenant quota usage high** alert is triggered.

  Consider asking your grid administrator to increase the quota.

- If you exceed your quota, a notification tells you that you can't upload new objects.

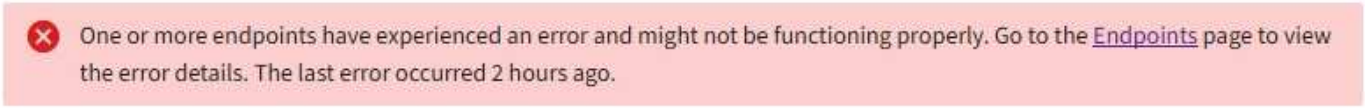
Capacity limit usage

If you've set a capacity limit for your buckets, the Tenant Manager dashboard displays a list of top buckets by capacity limit usage.

If no limit is set for a bucket, its capacity is unlimited. However, if your tenant account has a total storage quota and that quota is reached, you won't be able to ingest more objects regardless of the remaining capacity limit on a bucket.

Endpoint errors

If you have used the Grid Manager to configure one or more endpoints for use with platform services, the Tenant Manager dashboard displays an alert if any endpoint errors have occurred within the past seven days.



To see details about platform services endpoint errors, select **Endpoints** to display the Endpoints page.

Tenant Management API

Understand Tenant Management API

You can perform system management tasks using the Tenant Management REST API instead of the Tenant Manager user interface. For example, you might want to use the API to automate operations or to create multiple entities, such as users, more quickly.

The Tenant Management API:

- Uses the Swagger open source API platform. Swagger provides an intuitive user interface that allows developers and non-developers to interact with the API. The Swagger user interface provides complete details and documentation for each API operation.
- Uses versioning to support non-disruptive upgrades.

To access the Swagger documentation for the Tenant Management API:

1. Sign in to the Tenant Manager.
2. From the top of the Tenant Manager, select the help icon and select **API documentation**.

API operations

The Tenant Management API organizes the available API operations into the following sections:

- **account**: Operations on the current tenant account, including getting storage usage information.

- **auth**: Operations to perform user session authentication.

  The Tenant Management API supports the Bearer Token Authentication Scheme. For a tenant login, you provide a username, password, and accountId in the JSON body of the authentication request (that is, `POST /api/v3/authorize`). If the user is successfully authenticated, a security token is returned. This token must be provided in the header of subsequent API requests ("Authorization: Bearer token").

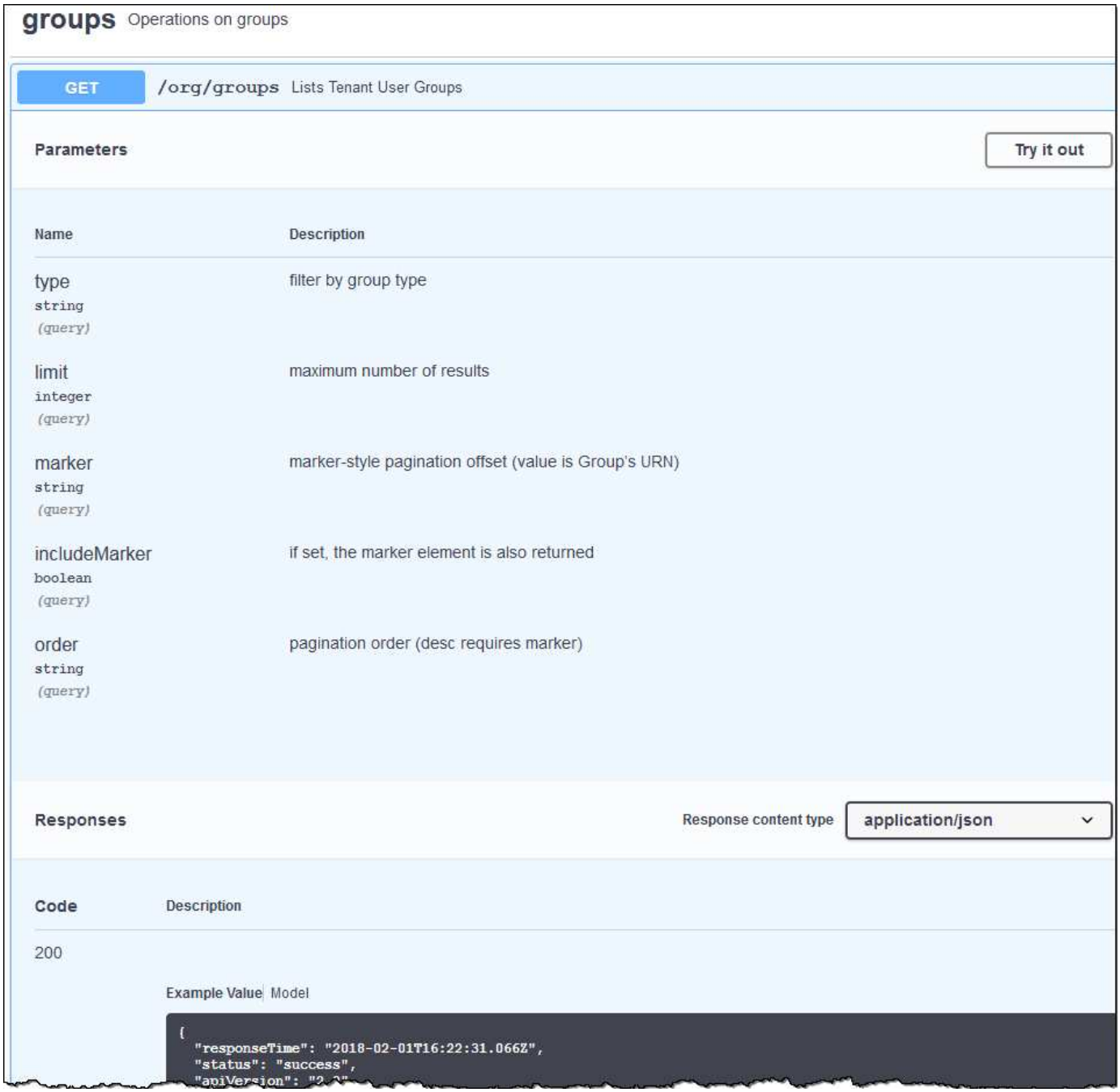
  For information about improving authentication security, see Protect against Cross-Site Request Forgery.

  > If single sign-on (SSO) is enabled for the StorageGRID system, you must perform different steps to authenticate. See the instructions for using the Grid Management API.

- **config**: Operations related to the product release and versions of the Tenant Management API. You can list the product release version and the major versions of the API supported by that release.
- **containers**: Operations on S3 buckets or Swift containers.
- **deactivated-features**: Operations to view features that might have been deactivated.
- **endpoints**: Operations to manage an endpoint. Endpoints allow an S3 bucket to use an external service for StorageGRID CloudMirror replication, notifications, or search integration.
- **grid-federation-connections**: Operations on grid federation connections and cross-grid replication.
- **groups**: Operations to manage local tenant groups and to retrieve federated tenant groups from an external identity source.
- **identity-source**: Operations to configure an external identity source and to manually synchronize federated group and user information.
- **ilm**: Operations on information lifecycle management (ILM) settings.
- **regions**: Operations to determine which regions have been configured for the StorageGRID system.
- **s3**: Operations to manage S3 access keys for tenant users.
- **s3-object-lock**: Operations on global S3 Object Lock settings, used to support regulatory compliance.
- **users**: Operations to view and manage tenant users.

Operation details

When you expand each API operation, you can see its HTTP action, endpoint URL, a list of any required or optional parameters, an example of the request body (when required), and the possible responses.



Issue API requests

Any API operations you perform using the API Documentation webpage are live operations. Be careful not to create, update, or delete configuration data or other data by mistake.

Steps

1. Select the HTTP action to see the request details.
2. Determine if the request requires additional parameters, such as a group or user ID. Then, obtain these values. You might need to issue a different API request first to get the information you need.
3. Determine if you need to modify the example request body. If so, you can select **Model** to learn the requirements for each field.
4. Select **Try it out**.

5. Provide any required parameters, or modify the request body as required.
6. Select **Execute**.
7. Review the response code to determine if the request was successful.

Tenant Management API versioning

The Tenant Management API uses versioning to support non-disruptive upgrades.

For example, this Request URL specifies version 4 of the API.

`https://hostname_or_ip_address/api/v4/authorize`

The major version of the API is bumped when changes are made that are *not compatible* with older versions. The minor version of the API is bumped when changes are made that *are compatible* with older versions. Compatible changes include the addition of new endpoints or new properties.

The following example illustrates how the API version is bumped based on the type of changes made.

Type of change to API	Old version	New version
Compatible with older versions	2.1	2.2
Not compatible with older versions	2.1	3.0
	3.0	4.0

When you install StorageGRID software for the first time, only the most recent version of the API is enabled. However, when you upgrade to a new feature release of StorageGRID, you continue to have access to the older API version for at least one StorageGRID feature release.

> You can configure the supported versions. See the **config** section of the Swagger API documentation for the Grid Management API for more information. You should deactivate support for the older version after updating all API clients to use the newer version.

Outdated requests are marked as deprecated in the following ways:

- The response header is "Deprecated: true"
- The JSON response body includes "deprecated": true
- A deprecated warning is added to nms.log. For example:

```
Received call to deprecated v2 API at POST "/api/v2/authorize"
```

Determine which API versions are supported in the current release

Use the `GET /versions` API request to return a list of the supported API major versions. This request is located in the **config** section of the Swagger API documentation.

```
GET https://{{IP-Address}}/api/versions
{
  "responseTime": "2023-06-27T22:13:50.750Z",
  "status": "success",
  "apiVersion": "4.0",
  "data": [
    2,
    3,
    4
  ]
}
```

Specify an API version for a request

You can specify the API version using a path parameter (`/api/v4`) or a header (`Api-Version: 4`). If you provide both values, the header value overrides the path value.

```
curl https://[IP-Address]/api/v4/grid/accounts

curl -H "Api-Version: 4" https://[IP-Address]/api/grid/accounts
```

Protect against Cross-Site Request Forgery (CSRF)

You can help protect against Cross-Site Request Forgery (CSRF) attacks against StorageGRID by using CSRF tokens to enhance authentication that uses cookies. The Grid Manager and Tenant Manager automatically enable this security feature; other API clients can choose whether to enable it when they sign in.

An attacker that can trigger a request to a different site (such as with an HTTP form POST) can cause certain requests to be made using the signed-in user's cookies.

StorageGRID helps protect against CSRF attacks by using CSRF tokens. When enabled, the contents of a specific cookie must match the contents of either a specific header or a specific POST body parameter.

To enable the feature, set the `csrfToken` parameter to `true` during authentication. The default is `false`.

```
curl -X POST --header "Content-Type: application/json" --header "Accept:
application/json" -d "{
  \"username\": \"MyUserName\",
  \"password\": \"MyPassword\",
  \"cookie\": true,
  \"csrfToken\": true
}" "https://example.com/api/v3/authorize"
```

When true, a `GridCsrfToken` cookie is set with a random value for sign-ins to the Grid Manager, and the `AccountCsrfToken` cookie is set with a random value for sign-ins to the Tenant Manager.

If the cookie is present, all requests that can modify the state of the system (POST, PUT, PATCH, DELETE) must include one of the following:

- The `X-Csrf-Token` header, with the value of the header set to the value of the CSRF token cookie.
- For endpoints that accept a form-encoded body: A `csrfToken` form-encoded request body parameter.

To configure CSRF protection, use the Grid Management API or Tenant Management API.

> Requests that have a CSRF token cookie set will also enforce the "Content-Type: application/json" header for any request that expects a JSON request body as an additional protection against CSRF attacks.

Use grid federation connections

Clone tenant groups and users

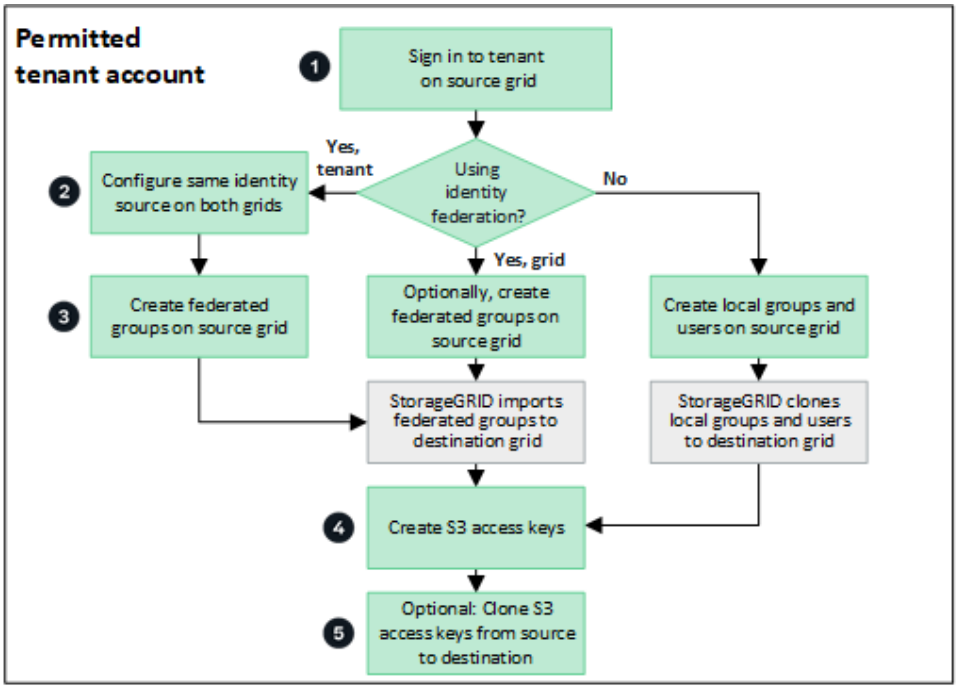
If a tenant was created or edited to use a grid federation connection, that tenant is replicated from one StorageGRID system (the source tenant) to another StorageGRID system (the replica tenant). After the tenant has been replicated, any groups and users added to the source tenant are cloned to the replica tenant.

The StorageGRID system where the tenant is originally created is the tenant's *source grid*. The StorageGRID system where the tenant is replicated is the tenant's *destination grid*. Both tenant accounts have the same account ID, name, description, storage quota, and assigned permissions, but the destination tenant does not initially have a root user password. For details, see What is account clone and Manage permitted tenants.

The cloning of tenant account information is required for cross-grid replication of bucket objects. Having the same tenant groups and users on both grids ensures you can access the corresponding buckets and objects on either grid.

Tenant workflow for account clone

If your tenant account has the **Use grid federation connection** permission, review the workflow diagram to see the steps you will perform to clone groups, users, and S3 access keys.



These are the primary steps in the workflow:

1 Sign in to tenant

Sign in to the tenant account on the source grid (the grid where the tenant was initially created.)

2 Optionally, configure identity federation

If your tenant account has the **Use own identity source** permission to use federated groups and users, configure the same identity source (with the same settings) for both the source and destination tenant accounts. Federated groups and users can't be cloned unless both grids are using the same identity source. For instructions, see Use identity federation.

3 Create groups and users

When creating groups and users, always start from the tenant's source grid. When you add a new group, StorageGRID automatically clones it to the destination grid.

- If identity federation is configured for the entire StorageGRID system or for your tenant account, create new tenant groups by importing federated groups from the identity source.
- If you aren't using identity federation, create new local groups and then create local users.

4 Create S3 access keys

You can create your own access keys or to create another user's access keys on either the source grid or the destination grid to access buckets on that grid.

5 Optionally, clone S3 access keys

If you need to access buckets with the same access keys on both grids, create the access keys on the source grid and then use the Tenant Manager API to manually clone them to the destination grid. For instructions, see Clone S3 access keys using the API.

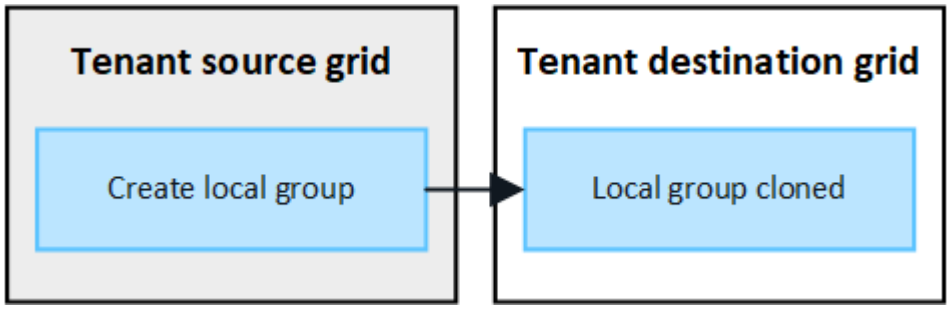
How are groups, users, and S3 access keys cloned?

Review this section to understand how groups, users, and S3 access keys are cloned between the tenant source grid and the tenant destination grid.

Local groups created on source grid are cloned

After a tenant account is created and replicated to the destination grid, StorageGRID automatically clones any local groups you add to the tenant's source grid to the tenant's destination grid.

Both the original group and its clone have the same access mode, group permissions, and S3 group policy. For instructions, see Create groups for S3 tenant.

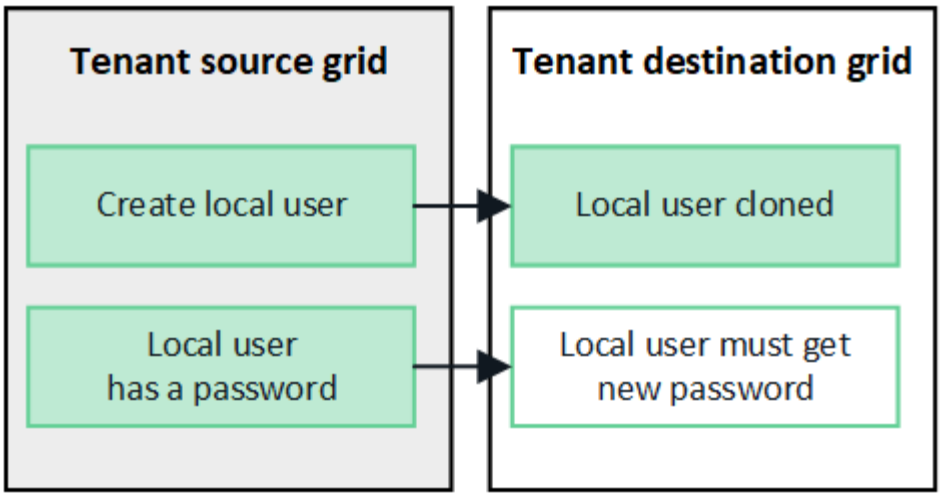


> Any users you select when you create a local group on the source grid aren't included when the group is cloned to the destination grid. For this reason, don't select users when you create the group. Instead, select the group when you create the users.

Local users created on source grid are cloned

When you create a new local user on the source grid, StorageGRID automatically clones that user to the destination grid. Both the original user and its clone have the same full name, username, and **Deny access** setting. Both users also belong to the same groups. For instructions, see Manage local users.

For security reasons, local user passwords aren't cloned to the destination grid. If a local user needs to access Tenant Manager on the destination grid, the root user for the tenant account must add a password for that user on the destination grid. For instructions, see Manage local users.

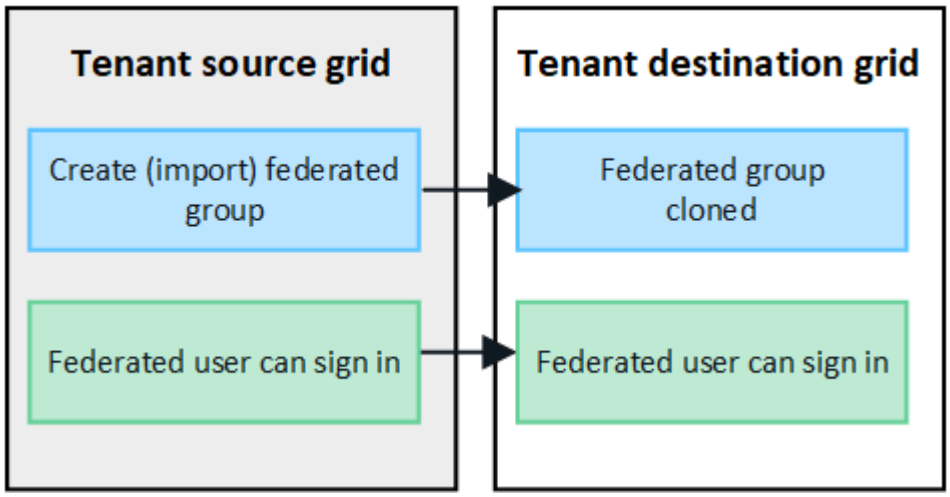


Federated groups created on source grid are cloned

Assuming the requirements for using account clone with single sign-on and identity federation have been met, federated groups that you create (import) for the tenant on the source grid are automatically cloned to the tenant on the destination grid.

Both groups have the same access mode, group permissions and S3 group policy.

After federated groups are created for the source tenant and cloned to the destination tenant, federated users can sign in to the tenant on either grid.

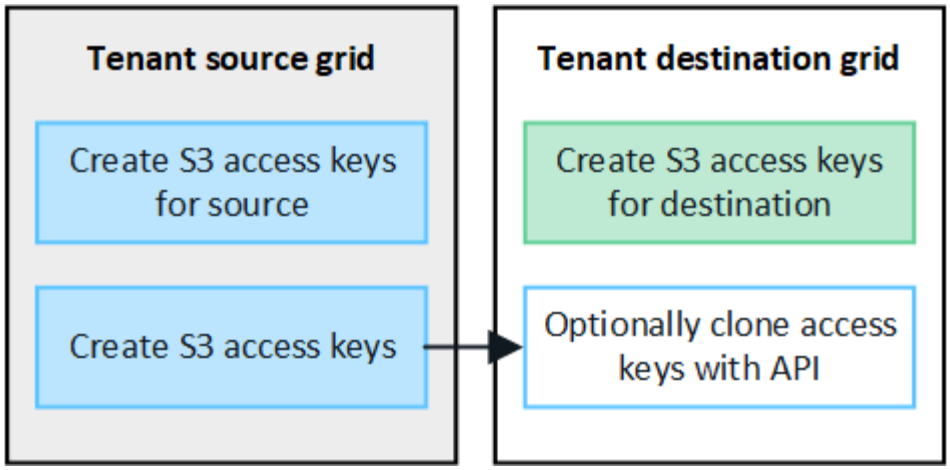


S3 access keys can be manually cloned

StorageGRID does not automatically clone S3 access keys because security is improved by having different keys on each grid.

To manage access keys on the two grids, you can do either of the following:

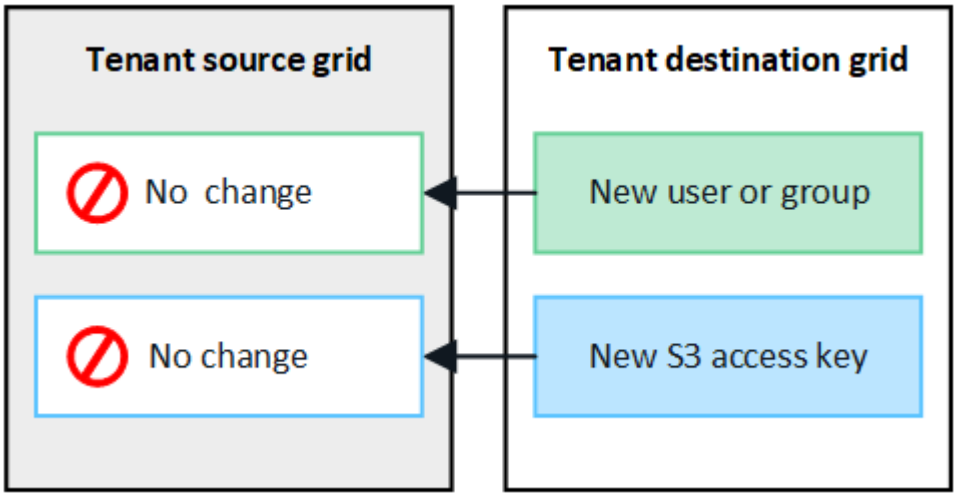
- If you don't need to use the same keys for each grid, you can create your own access keys or create another user's access keys on each grid.
- If you need to use the same keys on both grids, you can create keys on the source grid and then use the Tenant Manager API to manually clone the keys to the destination grid.



When you clone S3 access keys for a federated user, both the user and the S3 access keys are cloned to the destination tenant.

Groups and users added to destination grid aren't cloned

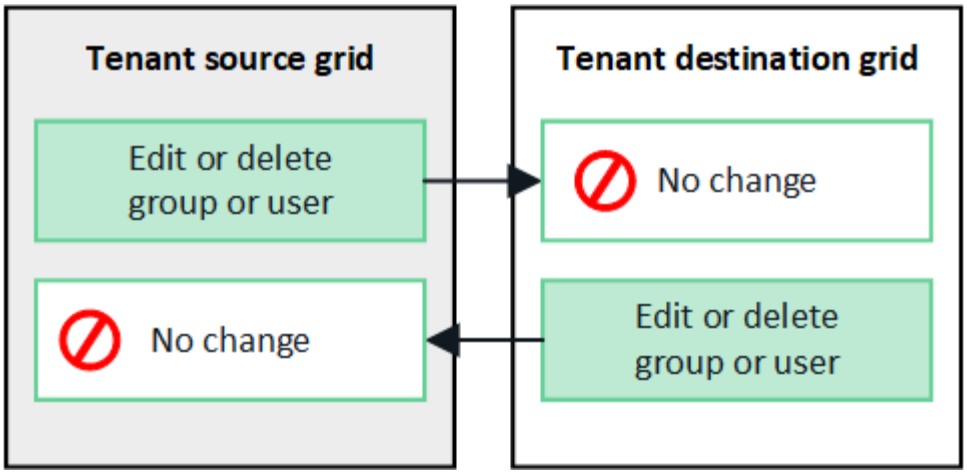
Cloning occurs only from the tenant's source grid to the tenant's destination grid. If you create or import groups and users on the tenant's destination grid, StorageGRID will not clone these items back the tenant's source grid.



Edited or deleted groups, users, and access keys aren't cloned

Cloning occurs only when you create new groups and users.

If you edit or delete groups, users, or access keys on either grid, your changes will not be cloned to the other grid.



Clone S3 access keys using the API

If your tenant account has the **Use grid federation connection** permission, you can use the Tenant Management API to manually clone S3 access keys from the tenant on the source grid to the tenant on the destination grid.

Before you begin

- The tenant account has the **Use grid federation connection** permission.
- The grid federation connection has a **Connection status** of **Connected**.
- You are signed in to the Tenant Manager on the tenant's source grid using a supported web browser.
- You belong to a user group that has the Manage your own S3 credentials or Root access permission.
- If you are cloning access keys for a local user, the user already exists on both grids.

> When you clone S3 access keys for a federated user, both the user and the S3 access keys are added to the destination tenant.

Clone your own access keys

You can clone your own access keys if you need to access the same buckets on both grids.

Steps

1. Using the Tenant Manager on the source grid, create your own access keys and download the `.csv` file.
2. From the top of the Tenant Manager, select the help icon and select **API documentation**.
3. In the **s3** section, select the following endpoint:

   ```
   POST /org/users/current-user/replicate-s3-access-key
   ```

4. Select **Try it out**.
5. In the **body** text box, replace the example entries for **accessKey** and **secretAccessKey** with the values from the **.csv** file you downloaded.

Be sure to retain the double quotes around each string.

6. If the key will expire, replace the example entry for **expires** with the expiration date and time as a string in ISO 8601 data-time format (for example, `2024-02-28T22:46:33-08:00`). If the key will not expire, enter **null** as the value for the **expires** entry (or remove the **Expires** line and the preceding comma).
7. Select **Execute**.
8. Confirm that the server response code is **204**, indicating that the key was successfully cloned to the destination grid.

Clone another user's access keys

You can clone another user's access keys if they need to access the same buckets on both grids.

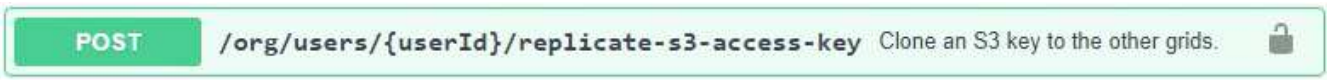
Steps

1. Using the Tenant Manager on the source grid, create the other user's S3 access keys and download the `.csv` file.
2. From the top of the Tenant Manager, select the help icon and select **API documentation**.
3. Obtain the user ID. You will need this value to clone the other user's access keys.
   a. From the **users** section, select the following endpoint:

      `GET /org/users`

   b. Select **Try it out**.
   c. Specify any parameters you want to use when looking up users.
   d. Select **Execute**.
   e. Find the user whose keys you want to clone, and copy the number in the **id** field.
4. In the **s3** section, select the following endpoint:

   `POST /org/users/{userId}/replicate-s3-access-key`

5. Select **Try it out**.
6. In the **userId** text box, paste the user ID you copied.
7. In the **body** text box, replace the example entries for **example access key** and **secret access key** with the values from the **.csv** file for that user.

   Be sure to retain the double quotes around the string.

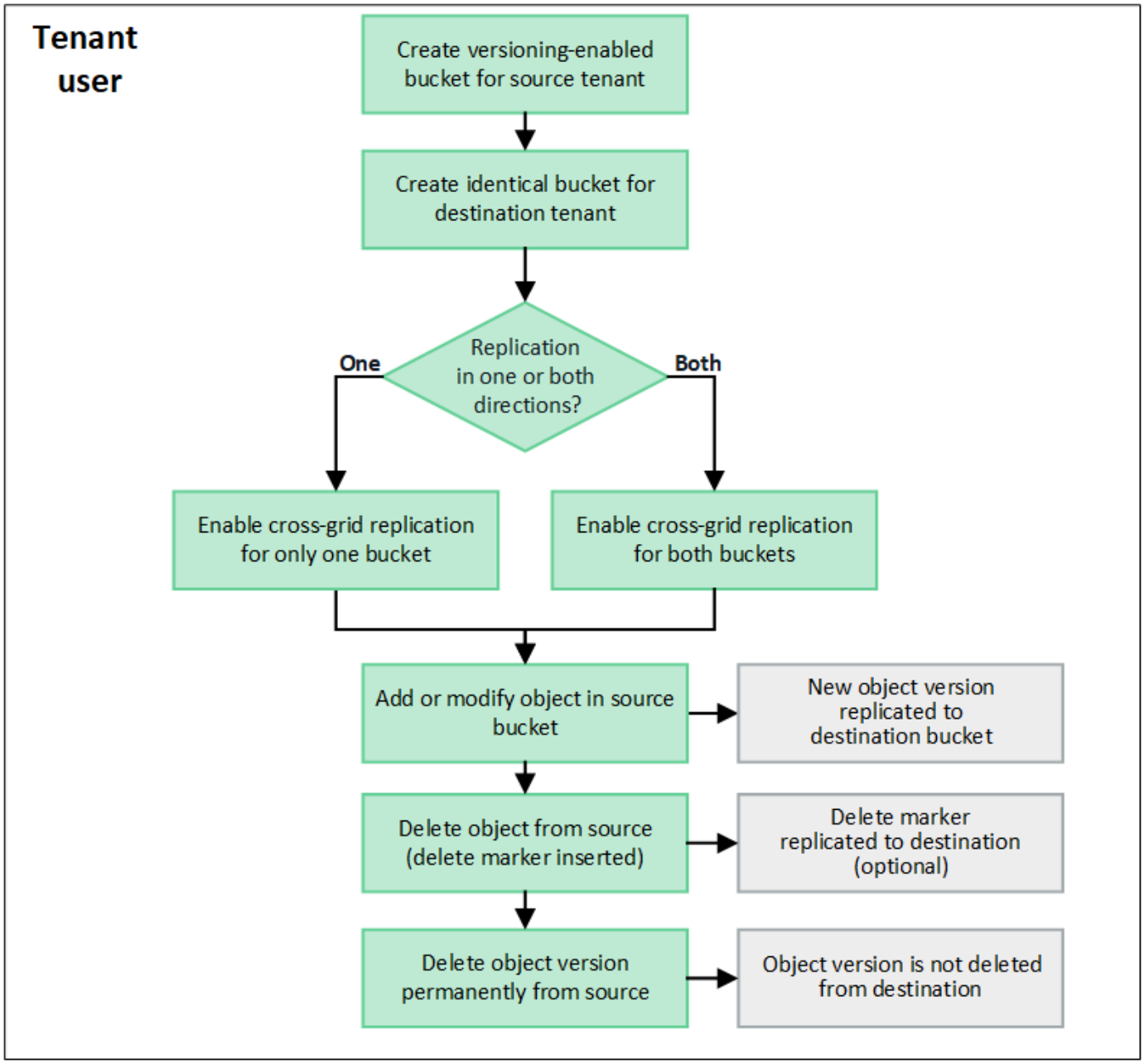
8. If the key will expire, replace the example entry for **expires** with the expiration date and time as a string in ISO 8601 data-time format (for example, `2023-02-28T22:46:33-08:00`). If the key will not expire, enter **null** as the value for the **expires** entry (or remove the **Expires** line and the preceding comma).
9. Select **Execute**.
10. Confirm that the server response code is **204**, indicating that the key was successfully cloned to the destination grid.

Manage cross-grid replication

If your tenant account was assigned the **Use grid federation connection** permission when it was created, you can use cross-grid replication to automatically replicate objects between buckets on the tenant's source grid and buckets on the tenant's destination grid. Cross-grid replication can occur in one or both directions.

Workflow for cross-grid replication

The workflow diagram summarize the steps you will perform to configure cross-grid replication between buckets on two grids. These steps are described in more detail below.



Configure cross-grid replication

Before you can use cross-grid replication, you must sign in to the corresponding tenant accounts on each grid and create identical buckets. Then, you can enable cross-grid replication on either or both buckets.

Before you begin

- You have reviewed the requirements for cross-grid replication. See What is cross-grid replication.
- You are using a supported web browser.
- The tenant account has the **Use grid federation connection** permission, and identical tenant accounts exist on both grids. See Manage the permitted tenants for grid federation connection.
- The tenant user you will be signing in as already exists on both grids and belongs to a user group that has the Root access permission.
- If you will be signing in to the tenant's destination grid as a local user, the root user for the tenant account has set a password for your user account on that grid.

Create two identical buckets

As a first step, sign in to the corresponding tenant accounts on each grid and create identical buckets.

Steps

1. Starting from either grid in the grid federation connection, create a new bucket:
   a. Sign in to the tenant account using the credentials of a tenant user who exists on both grids.

      If you are unable to sign in to the tenant's destination grid as a local user, confirm that the root user for the tenant account has set a password for your user account.

   b. Follow the instructions to create an S3 bucket.
   c. On the **Manage object settings** tab, select **Enable object versioning**.
   d. If S3 Object Lock is enabled for your StorageGRID system, don't enable S3 Object Lock for the bucket.
   e. Select **Create bucket**.
   f. Select **Finish**.
2. Repeat these steps to create an identical bucket for the same tenant account on the other grid in the grid federation connection.

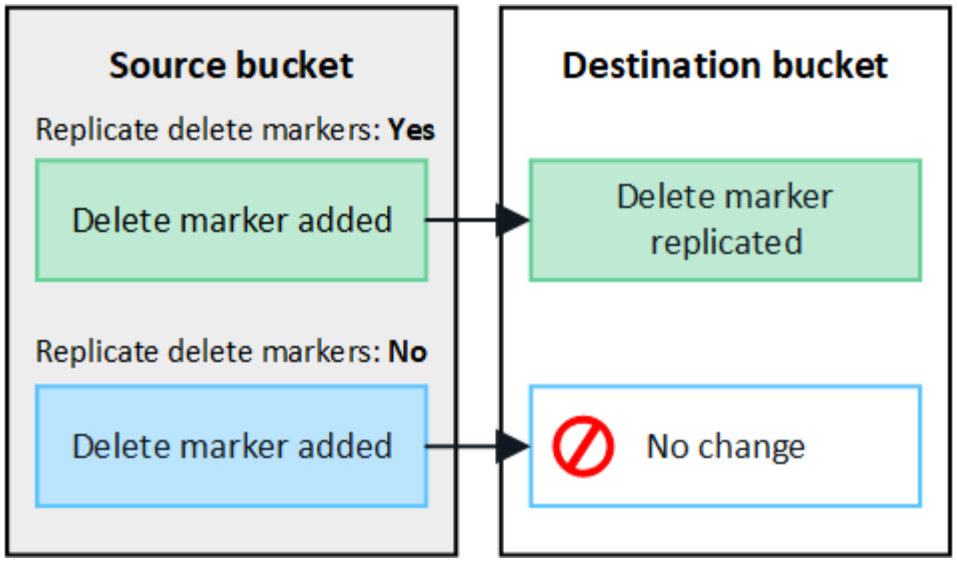
   As required, each bucket can use a different region.

Enable cross-grid replication

You must perform these steps before adding any objects to either bucket.

Steps

1. Starting from a grid whose objects you want to replicate, enable cross-grid replication in one direction:
   a. Sign in to the tenant account for the bucket.
   b. Select **View buckets** from the dashboard, or select **STORAGE (S3)** > **Buckets**.
   c. Select the bucket name from the table to access the bucket details page.
   d. Select the **Cross-grid replication** tab.
   e. Select **Enable**, and review the list of requirements.
   f. If all requirements have been met, select the grid federation connection you want to use.
   g. Optionally, change the setting of **Replicate delete markers** to determine what happens on the destination grid if an S3 client issues a delete request to the source grid that doesn't include a version ID:
      - **Yes** (default): A delete marker is added to the source bucket and replicated to the destination bucket.
      - **No**: A delete marker is added to the source bucket but is not replicated to the destination bucket.



> If the delete request includes a version ID, that object version is permanently removed from the source bucket. StorageGRID does not replicate delete requests that include a version ID, so the same object version is not deleted from the destination.

See What is cross-grid replication for details.

h. Optionally, change the setting of the **Cross-grid replication** audit category to manage the volume of audit messages:
   - **Error** (default): Only failed cross-grid replication requests are included in the audit output.
   - **Normal**: All cross-grid replication requests are included, which significantly increases the volume of the audit output.

i. Review your selections. You aren't able to change these settings unless both buckets are empty.

j. Select **Enable and test**.

   After a few moments, a success message appears. Objects added to this bucket will now be automatically replicated to the other grid. **Cross-grid replication** is shown as an enabled feature on the bucket details page.

2. Optionally, go to the corresponding bucket on the other grid and enable cross-grid replication in both directions.

Test replication between grids

If cross-grid replication is enabled for a bucket, you might need to verify that the connection and cross-grid replication are working correctly and that the source and destination buckets still meet all requirements (for example, versioning is still enabled).

Before you begin

- You are using a supported web browser.
- You belong to a user group that has the Root access permission.

Steps

1. Sign in to the tenant account for the bucket.

2. Select **View buckets** from the dashboard, or select **STORAGE (S3)** > **Buckets**.
3. Select the bucket name from the table to access the bucket details page.
4. Select the **Cross-grid replication** tab.
5. Select **Test connection**.

   If the connection is healthy, a success banner appears. Otherwise, an error message appears, which you and the grid admin can use to resolve the issue. For details, see Troubleshoot grid federation errors.

6. If cross-grid replication is configured to occur in both directions, go to the corresponding bucket on the other grid and select **Test connection** to verify that cross-grid replication is working in the other direction.

Disable cross-grid replication

You can permanently stop cross-grid replication if you no longer want to copy objects to the other grid.

Before disabling cross-grid replication, note the following:

- Disabling cross-grid replication does not remove any objects that have already been copied between grids. For example, objects in `my-bucket` on Grid 1 that have been copied to `my-bucket` on Grid 2 aren't removed if you disable cross-grid replication for that bucket. If you want to delete these objects, you must remove them manually.
- If cross-grid replication was enabled for each of the buckets (that is, if replication occurs in both directions), you can disable cross-grid replication for either or both buckets. For example, you might want to disable replicating objects from `my-bucket` on Grid 1 to `my-bucket` on Grid 2, while continuing to replicate objects from `my-bucket` on Grid 2 to `my-bucket` on Grid 1.
- You must disable cross-grid replication before you can remove a tenant's permission to use the grid federation connection. See Manage permitted tenants.
- If you disable cross-grid replication for a bucket that contains objects, you will not be able to reenable cross-grid replication unless you delete all objects from both the source and destination buckets.

  You can't reenable replication unless both buckets are empty.

Before you begin

- You are using a supported web browser.
- You belong to a user group that has the Root access permission.

Steps

1. Starting from the grid whose objects you no longer want to replicate, stop cross-grid replication for the bucket:
   a. Sign in to the tenant account for the bucket.
   b. Select **View buckets** from the dashboard, or select **STORAGE (S3)** > **Buckets**.
   c. Select the bucket name from the table to access the bucket details page.
   d. Select the **Cross-grid replication** tab.
   e. Select **Disable replication**.
   f. If you are sure you want to disable cross-grid replication for this bucket, type **Yes** in the text box, and select **Disable**.

After a few moments, a success message appears. New objects added to this bucket can no longer be automatically replicated to the other grid. **Cross-grid replication** is no longer shown as a Enabled feature on the Buckets page.

2. If cross-grid replication was configured to occur in both directions, go to the corresponding bucket on the other grid and stop cross-grid replication in the other direction.

View grid federation connections

If your tenant account has the **Use grid federation connection** permission, you can view the allowed connections.

Before you begin

- The tenant account has the **Use grid federation connection** permission.
- You are signed in to the Tenant Manager using a supported web browser.
- You belong to a user group that has the Root access permission.

Steps

1. Select **STORAGE (S3)** > **Grid federation connections**.

   The Grid federation connection page appears and includes a table that summarizes the following information:

Column	Description
Connection name	The grid federation connections this tenant has permission to use.
Buckets with cross-grid replication	For each grid federation connection, the tenant buckets that have cross-grid replication enabled. Objects added to these buckets will be replicated to the other grid in the connection.
Last error	For each grid federation connection, the most recent error to occur, if any, when data was being replicated to the other grid. See Clear the last error.

2. Optionally, select a bucket name to view bucket details.

Clear the last error

An error might appear in the **Last error** column for one of these reasons:

- The source object version was not found.
- The source bucket was not found.
- The destination bucket was deleted.
- The destination bucket was re-created by a different account.
- The destination bucket has versioning suspended.
- The destination bucket was re-created by the same account but is now unversioned.

This column only shows the last cross-grid replication error to occur; previous errors that might have occurred will not be shown.

Steps

1. If a message appears in the **Last error** column, view the message text.

   For example, this error indicates that the destination bucket for cross-grid replication was in an invalid state, possibly because versioning was suspended or S3 Object Lock was enabled.

2. Perform any recommended actions. For example, if versioning was suspended on the destination bucket for cross-grid replication, reenable versioning for that bucket.
3. Select the connection from the table.
4. Select **Clear error**.
5. Select **Yes** to clear the message and update the system's status.
6. Wait 5-6 minutes and then ingest a new object into the bucket. Confirm that the error message does not reappear.

   To ensure the error message is cleared, wait at least 5 minutes after the timestamp in the message before ingesting a new object.

7. To determine if any objects failed to be replicated because of the bucket error, see Identify and retry failed replication operations.

Manage groups and users

Use identity federation

Using identity federation makes setting up tenant groups and users faster, and it allows tenant users to sign in to the tenant account using familiar credentials.

Configure identity federation for Tenant Manager

You can configure identity federation for the Tenant Manager if you want tenant groups and users to be managed in another system such as Active Directory, Azure Active Directory (Azure AD), OpenLDAP, or Oracle Directory Server.

Before you begin

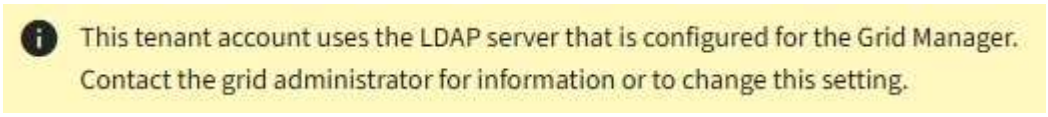
- You are signed in to the Tenant Manager using a supported web browser.
- You belong to a user group that has the Root access permission.
- You are using Active Directory, Azure AD, OpenLDAP, or Oracle Directory Server as the identity provider.

  If you want to use an LDAP v3 service that is not listed, contact technical support.

- If you plan to use OpenLDAP, you must configure the OpenLDAP server. See Guidelines for configuring OpenLDAP server.
- If you plan to use Transport Layer Security (TLS) for communications with the LDAP server, the identity provider must be using TLS 1.2 or 1.3. See Supported ciphers for outgoing TLS connections.

About this task

Whether you can configure an identity federation service for your tenant depends on how your tenant account was set up. Your tenant might share the identity federation service that was configured for the Grid Manager. If you see this message when you access the Identity Federation page, you can't configure a separate federated identity source for this tenant.

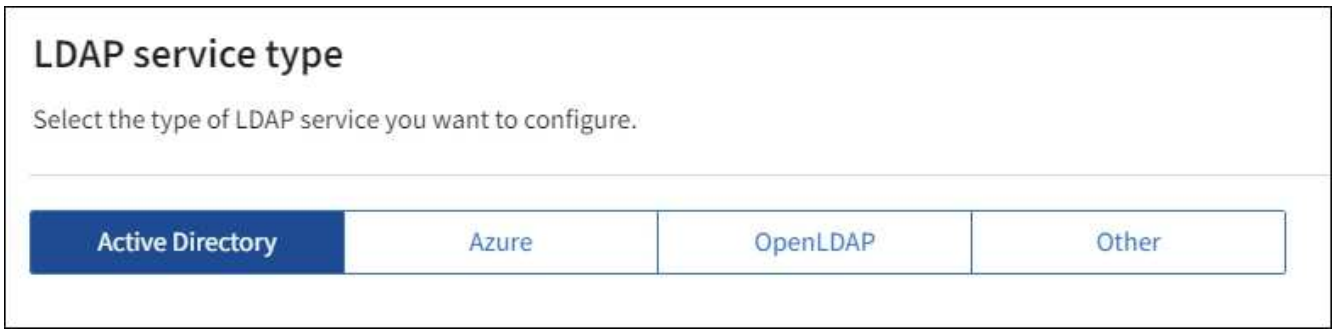


Enter configuration

When you configure identify federation, you provide the values StorageGRID needs to connect to an LDAP service.

Steps

1. Select **ACCESS MANAGEMENT** > **Identity federation**.
2. Select **Enable identity federation**.
3. In the LDAP service type section, select the type of LDAP service you want to configure.



   Select **Other** to configure values for an LDAP server that uses Oracle Directory Server.

4. If you selected **Other**, complete the fields in the LDAP Attributes section. Otherwise, go to the next step.
   - **User Unique Name**: The name of the attribute that contains the unique identifier of an LDAP user. This attribute is equivalent to `sAMAccountName` for Active Directory and `uid` for OpenLDAP. If you are configuring Oracle Directory Server, enter `uid`.
   - **User UUID**: The name of the attribute that contains the permanent unique identifier of an LDAP user. This attribute is equivalent to `objectGUID` for Active Directory and `entryUUID` for OpenLDAP. If you

are configuring Oracle Directory Server, enter `nsuniqueid`. Each user's value for the specified attribute must be a 32-digit hexadecimal number in either 16-byte or string format, where hyphens are ignored.

- **Group Unique Name**: The name of the attribute that contains the unique identifier of an LDAP group. This attribute is equivalent to `sAMAccountName` for Active Directory and `cn` for OpenLDAP. If you are configuring Oracle Directory Server, enter `cn`.
- **Group UUID**: The name of the attribute that contains the permanent unique identifier of an LDAP group. This attribute is equivalent to `objectGUID` for Active Directory and `entryUUID` for OpenLDAP. If you are configuring Oracle Directory Server, enter `nsuniqueid`. Each group's value for the specified attribute must be a 32-digit hexadecimal number in either 16-byte or string format, where hyphens are ignored.

5. For all LDAP service types, enter the required LDAP server and network connection information in the Configure LDAP server section.
   - **Hostname**: The fully qualified domain name (FQDN) or IP address of the LDAP server.
   - **Port**: The port used to connect to the LDAP server.

     > The default port for STARTTLS is 389, and the default port for LDAPS is 636. However, you can use any port as long as your firewall is configured correctly.

   - **Username**: The full path of the distinguished name (DN) for the user that will connect to the LDAP server.

     For Active Directory, you can also specify the Down-Level Logon Name or the User Principal Name.

     The specified user must have permission to list groups and users and to access the following attributes:

     - `sAMAccountName` or `uid`
     - `objectGUID`, `entryUUID`, or `nsuniqueid`
     - `cn`
     - `memberOf` or `isMemberOf`
     - **Active Directory**: `objectSid`, `primaryGroupID`, `userAccountControl`, and `userPrincipalName`
     - **Azure**: `accountEnabled` and `userPrincipalName`

   - **Password**: The password associated with the username.

     > If you change the password in the future, you must update it on this page.

   - **Group Base DN**: The full path of the distinguished name (DN) for an LDAP subtree you want to search for groups. In the Active Directory example (below), all groups whose Distinguished Name is relative to the base DN (DC=storagegrid,DC=example,DC=com) can be used as federated groups.

     > The **Group unique name** values must be unique within the **Group Base DN** they belong to.

   - **User Base DN**: The full path of the distinguished name (DN) of an LDAP subtree you want to search for users.

> The **User unique name** values must be unique within the **User Base DN** they belong to.

- **Bind username format** (optional): The default username pattern StorageGRID should use if the pattern can't be determined automatically.

  Providing **Bind username format** is recommended because it can allow users to sign in if StorageGRID is unable to bind with the service account.

  Enter one of these patterns:

  - **UserPrincipalName pattern (Active Directory and Azure)**: `[USERNAME]@example.com`
  - **Down-level logon name pattern (Active Directory and Azure)**: `example\[USERNAME]`
  - **Distinguished name pattern**: `CN=[USERNAME],CN=Users,DC=example,DC=com`

    Include **[USERNAME]** exactly as written.

6. In the Transport Layer Security (TLS) section, select a security setting.
   - **Use STARTTLS**: Use STARTTLS to secure communications with the LDAP server. This is the recommended option for Active Directory, OpenLDAP, or Other, but this option is not supported for Azure.
   - **Use LDAPS**: The LDAPS (LDAP over SSL) option uses TLS to establish a connection to the LDAP server. You must select this option for Azure.
   - **Do not use TLS**: The network traffic between the StorageGRID system and the LDAP server will not be secured. This option is not supported for Azure.

     > Using the **Do not use TLS** option is not supported if your Active Directory server enforces LDAP signing. You must use STARTTLS or LDAPS.

7. If you selected STARTTLS or LDAPS, choose the certificate used to secure the connection.
   - **Use operating system CA certificate**: Use the default Grid CA certificate installed on the operating system to secure connections.
   - **Use custom CA certificate**: Use a custom security certificate.

     If you select this setting, copy and paste the custom security certificate into the CA certificate text box.

Test the connection and save the configuration

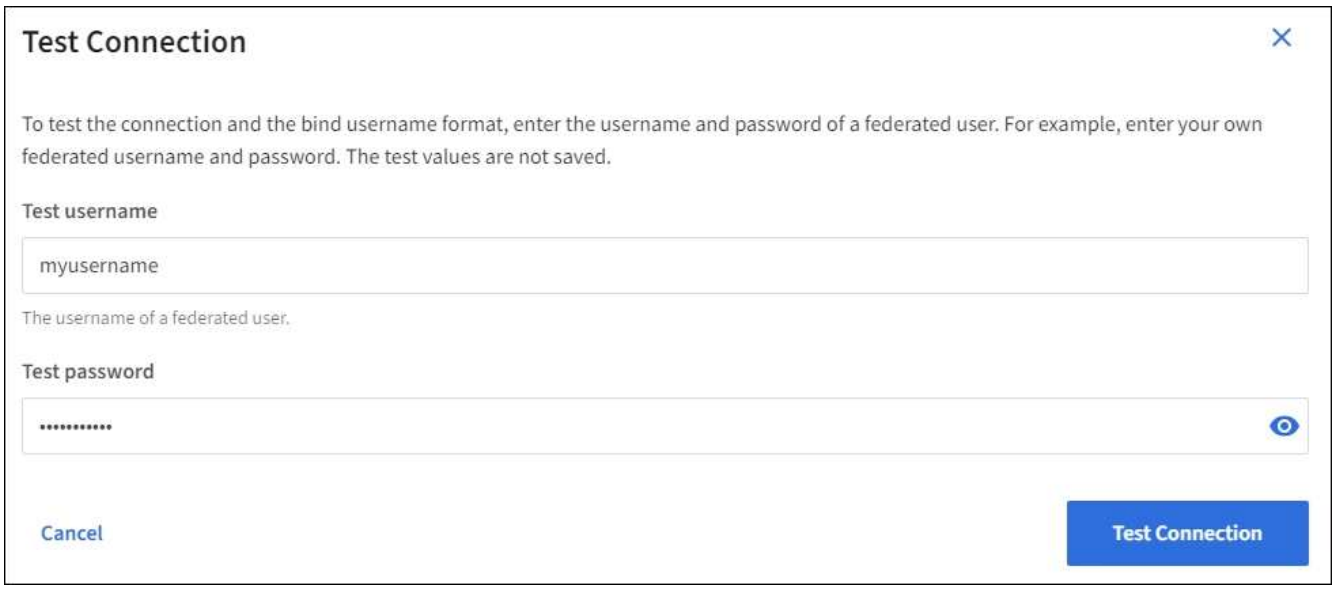
After entering all values, you must test the connection before you can save the configuration. StorageGRID verifies the connection settings for the LDAP server and the bind username format, if you provided one.

Steps

1. Select **Test connection**.
2. If you did not provide a bind username format:
   - A "Test connection successful" message appears if the connection settings are valid. Select **Save** to save the configuration.
   - A "test connection could not be established" message appears if the connection settings are invalid. Select **Close**. Then, resolve any issues and test the connection again.

3. If you provided a bind username format, enter the username and password of a valid federated user.

   For example, enter your own username and password. Don't include any special characters in the username, such as @ or /.



   - A "Test connection successful" message appears if the connection settings are valid. Select **Save** to save the configuration.
   - An error message appears if the connection settings, bind username format, or test username and password are invalid. Resolve any issues and test the connection again.

Force synchronization with identity source

The StorageGRID system periodically synchronizes federated groups and users from the identity source. You can force synchronization to start if you want to enable or restrict user permissions as quickly as possible.

Steps

1. Go to the Identity federation page.
2. Select **Sync server** at the top of the page.

   The synchronization process might take some time depending on your environment.

   > The **Identity federation synchronization failure** alert is triggered if there is an issue synchronizing federated groups and users from the identity source.

Disable identity federation

You can temporarily or permanently disable identity federation for groups and users. When identity federation is disabled, there is no communication between StorageGRID and the identity source. However, any settings you have configured are retained, allowing you to easily reenable identity federation in the future.

About this task

Before you disable identity federation, you should be aware of the following:

- Federated users will be unable to sign in.

- Federated users who are currently signed in will retain access to the StorageGRID system until their session expires, but they will be unable to sign in after their session expires.
- Synchronization between the StorageGRID system and the identity source will not occur, and alerts will not be raised for accounts that have not been synchronized.
- The **Enable identity federation** checkbox is disabled if single sign-on (SSO) is set to **Enabled** or **Sandbox Mode**. The SSO Status on the Single Sign-on page must be **Disabled** before you can disable identity federation. See Disable single sign-on.

Steps

1. Go to the Identity federation page.
2. Uncheck the **Enable identity federation** checkbox.

Guidelines for configuring OpenLDAP server

If you want to use an OpenLDAP server for identity federation, you must configure specific settings on the OpenLDAP server.

> For identity sources that aren't ActiveDirectory or Azure, StorageGRID will not automatically block S3 access to users who are disabled externally. To block S3 access, delete any S3 keys for the user or remove the user from all groups.

Memberof and refint overlays

The memberof and refint overlays should be enabled. For more information, see the instructions for reverse group membership maintenance in the OpenLDAP documentation: Version 2.4 Administrator's Guide.

Indexing

You must configure the following OpenLDAP attributes with the specified index keywords:

- `olcDbIndex: objectClass eq`
- `olcDbIndex: uid eq,pres,sub`
- `olcDbIndex: cn eq,pres,sub`
- `olcDbIndex: entryUUID eq`

In addition, ensure the fields mentioned in the help for Username are indexed for optimal performance.

See the information about reverse group membership maintenance in the OpenLDAP documentation: Version 2.4 Administrator's Guide.

Manage tenant groups

Create groups for an S3 tenant

You can manage permissions for S3 user groups by importing federated groups or creating local groups.

Before you begin

- You are signed in to the Tenant Manager using a supported web browser.

- You belong to a user group that has the Root access permission.
- If you plan to import a federated group, you have configured identity federation, and the federated group already exists in the configured identity source.
- If your tenant account has the **Use grid federation connection** permission, you have reviewed the workflow and considerations for cloning tenant groups and users, and you are signed in to the tenant's source grid.

Access the Create group wizard

As your first step, access the Create group wizard.

Steps

1. Select **ACCESS MANAGEMENT** > **Groups**.
2. If your tenant account has the **Use grid federation connection** permission, confirm that a blue banner appears, indicating that new groups created on this grid will be cloned to the same tenant on the other grid in the connection. If this banner does not appear, you might be signed in to the tenant's destination grid.

3. Select **Create group**.

Choose a group type

You can create a local group or import a federated group.

Steps

1. Select the **Local group** tab to create a local group, or select the **Federated group** tab to import a group from the previously configured identity source.

   If single sign-on (SSO) is enabled for your StorageGRID system, users belonging to local groups will not be able to sign in to the Tenant Manager, although they can use client applications to manage the tenant's resources, based on group permissions.

2. Enter the group's name.
   - **Local group**: Enter both a display name and a unique name. You can edit the display name later.

     If your tenant account has the **Use grid federation connection** permission, a cloning error will occur if the same **Unique name** already exists for the tenant on the destination grid.

- **Federated group**: Enter the unique name. For Active Directory, the unique name is the name associated with the `sAMAccountName` attribute. For OpenLDAP, the unique name is the name associated with the `uid` attribute.

3. Select **Continue**.

Manage group permissions

Group permissions control which tasks users can perform in the Tenant Manager and Tenant Management API.

Steps

1. For **Access mode**, select one of the following:
   - **Read-write** (default): Users can sign in to Tenant Manager and manage the tenant configuration.
   - **Read-only**: Users can only view settings and features. They can't make any changes or perform any operations in the Tenant Manager or Tenant Management API. Local read-only users can change their own passwords.

   > If a user belongs to multiple groups and any group is set to Read-only, the user will have read-only access to all selected settings and features.

2. Select one or more permissions for this group.

   See Tenant management permissions.

3. Select **Continue**.

Set S3 group policy

The group policy determines which S3 access permissions users will have.

Steps

1. Select the policy you want to use for this group.

Group policy	Description
No S3 Access	Default. Users in this group don't have access to S3 resources, unless access is granted with a bucket policy. If you select this option, only the root user will have access to S3 resources by default.
Read Only Access	Users in this group have read-only access to S3 resources. For example, users in this group can list objects and read object data, metadata, and tags. When you select this option, the JSON string for a read-only group policy appears in the text box. You can't edit this string.
Full Access	Users in this group have full access to S3 resources, including buckets. When you select this option, the JSON string for a full-access group policy appears in the text box. You can't edit this string.

Group policy	Description
Ransomware Mitigation	This example policy applies to all buckets for this tenant. Users in this group can perform common actions, but can't permanently delete objects from buckets that have object versioning enabled. Tenant Manager users who have the **Manage all buckets** permission can override this group policy. Limit the Manage all buckets permission to trusted users, and use Multi-Factor Authentication (MFA) where available.
Custom	Users in the group are granted the permissions you specify in the text box.

2. If you selected **Custom**, enter the group policy. Each group policy has a size limit of 5,120 bytes. You must enter a valid JSON formatted string.

   For detailed information about group policies, including language syntax and examples, see Example group policies.

3. If you are creating a local group, select **Continue**. If you are creating a federated group, select **Create group** and **Finish**.

Add users (local groups only)

You can save the group without adding users, or you can optionally add any local users that already exist.

> If your tenant account has the **Use grid federation connection** permission, any users you select when you create a local group on the source grid aren't included when the group is cloned to the destination grid. For this reason, don't select users when you create the group. Instead, select the group when you create the users.

Steps

1. Optionally, select one or more local users for this group.
2. Select **Create group** and **Finish**.

   The group you created appears in the list of groups.

   If your tenant account has the **Use grid federation connection** permission and you are on the tenant's source grid, the new group is cloned to the tenant's destination grid. **Success** appears as the **Cloning status** in the Overview section of the group's detail page.

Create groups for a Swift tenant

You can manage access permissions for a Swift tenant account by importing federated groups or creating local groups. At least one group must have the Swift Administrator permission, which is required to manage the containers and objects for a Swift tenant account.

Support for Swift client applications has been deprecated and will be removed in a future release.

Before you begin

- You are signed in to the Tenant Manager using a supported web browser.
- You belong to a user group that has the Root access permission.
- If you plan to import a federated group, you have configured identity federation, and the federated group already exists in the configured identity source.

Access the Create group wizard

Steps

As your first step, access the Create group wizard.

1. Select **ACCESS MANAGEMENT** > **Groups**.
2. Select **Create group**.

Choose a group type

You can create a local group or import a federated group.

Steps

1. Select the **Local group** tab to create a local group, or select the **Federated group** tab to import a group from the previously configured identity source.

   If single sign-on (SSO) is enabled for your StorageGRID system, users belonging to local groups will not be able to sign in to the Tenant Manager, although they can use client applications to manage the tenant's resources, based on group permissions.

2. Enter the group's name.
   - **Local group**: Enter both a display name and a unique name. You can edit the display name later.
   - **Federated group**: Enter the unique name. For Active Directory, the unique name is the name associated with the `sAMAccountName` attribute. For OpenLDAP, the unique name is the name associated with the `uid` attribute.
3. Select **Continue**.

Manage group permissions

Group permissions control which tasks users can perform in the Tenant Manager and Tenant Management API.

Steps

1. For **Access mode**, select one of the following:
   - **Read-write** (default): Users can sign in to Tenant Manager and manage the tenant configuration.
   - **Read-only**: Users can only view settings and features. They can't make any changes or perform any operations in the Tenant Manager or Tenant Management API. Local read-only users can change their own passwords.

If a user belongs to multiple groups and any group is set to Read-only, the user will have read-only access to all selected settings and features.

2. Select the **Root access** checkbox if group users need to sign in to the Tenant Manager or Tenant Management API.
3. Select **Continue**.

Set Swift group policy

Swift users need administrator permission to authenticate into the Swift REST API to create containers and ingest objects.

1. Select the **Swift administrator** checkbox if group users need to use the Swift REST API to manage containers and objects.
2. If you are creating a local group, select **Continue**. If you are creating a federated group, select **Create group** and **Finish**.

Add users (local groups only)

You can save the group without adding users, or you can optionally add any local users that already exist.

Steps

1. Optionally, select one or more local users for this group.

   If you have not yet created local users, you can add this group to the user on the Users page. See Manage local users.

2. Select **Create group** and **Finish**.

   The group you created appears in the list of groups.

Tenant management permissions

Before you create a tenant group, consider which permissions you want to assign to that group. Tenant management permissions determine which tasks users can perform using the Tenant Manager or the Tenant Management API. A user can belong to one or more groups. Permissions are cumulative if a user belongs to multiple groups.

To sign in to the Tenant Manager or to use the Tenant Management API, users must belong to a group that has at least one permission. All users who can sign in can perform the following tasks:

- View the dashboard
- Change their own password (for local users)

For all permissions, the group's Access mode setting determines whether users can change settings and perform operations or whether they can only view the related settings and features.

If a user belongs to multiple groups and any group is set to Read-only, the user will have read-only access to all selected settings and features.

You can assign the following permissions to a group. Note that S3 tenants and Swift tenants have different

group permissions.

Permission	Description	Details
Root access	Provides full access to the Tenant Manager and the Tenant Management API.	Swift users must have Root access permission to sign in to the tenant account.
Administrator	Swift tenants only. Provides full access to the Swift containers and objects for this tenant account	Swift users must have the Swift Administrator permission to perform any operations with the Swift REST API.
Manage your own S3 credentials	Allows users to create and remove their own S3 access keys.	Users who don't have this permission don't see the **STORAGE (S3)** > **My S3 access keys** menu option.
View all buckets	**S3 tenants**: Allows users to view all buckets and bucket configurations. **Swift tenants**: Allows Swift users to view all containers and container configurations using the Tenant Management API.	Users who don't have either the View all buckets or the Manage all buckets permission don't see the **Buckets** menu option. This permission is superseded by the Manage all buckets permission. It does not affect S3 bucket or group polices used by S3 clients or S3 Console. You can only assign this permission to Swift groups from the Tenant Management API. You can't assign this permission to Swift groups using the Tenant Manager.
Manage all buckets	**S3 tenants**: Allows users to use the Tenant Manager and the Tenant Management API to create and delete S3 buckets and to manage the settings for all S3 buckets in the tenant account, regardless of S3 bucket or group policies. **Swift tenants**: Allows Swift users to control the consistency for Swift containers using the Tenant Management API.	Users who don't have either the View all buckets or the Manage all buckets permission don't see the **Buckets** menu option. This permission supersedes the View all buckets permission. It does not affect S3 bucket or group polices used by S3 clients or S3 Console. You can only assign this permission to Swift groups from the Tenant Management API. You can't assign this permission to Swift groups using the Tenant Manager.
Manage endpoints	Allows users to use the Tenant Manager or the Tenant Management API to create or edit platform service endpoints, which are used as the destination for StorageGRID platform services.	Users who don't have this permission don't see the **Platform services endpoints** menu option.

Permission	Description	Details
Use S3 Console tab	When combined with the View all buckets or Manage all buckets permission, allows users to view and manage objects from the S3 Console tab on the details page for a bucket.	

Manage groups

Manage your tenant groups as needed to view, edit, or duplicate a group, and more.

Before you begin

- You are signed in to the Tenant Manager using a supported web browser.
- You belong to a user group that has the Root access permission.

View or edit group

You can view and edit the basic information and details for each group.

Steps

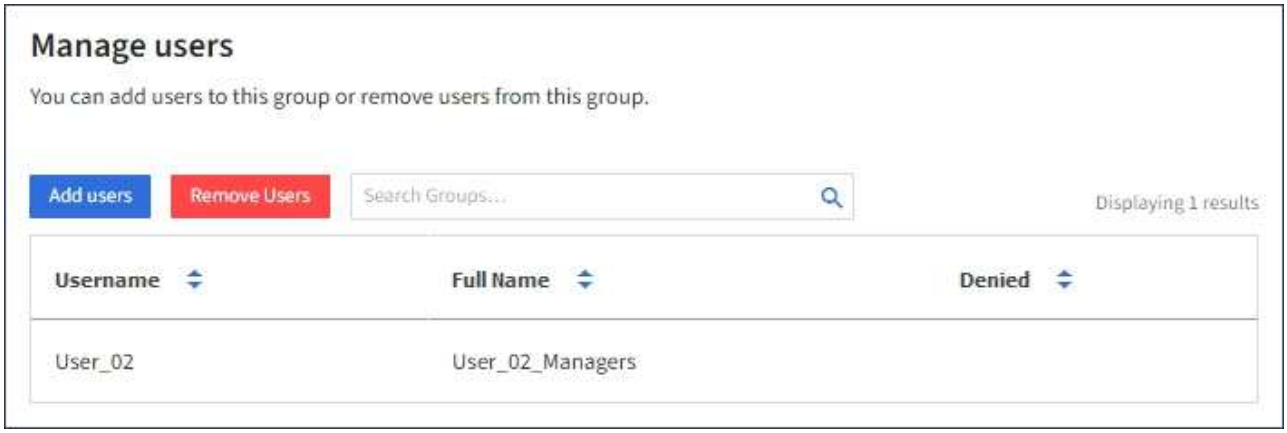
1. Select **ACCESS MANAGEMENT** > **Groups**.
2. Review the information provided on the Groups page, which lists basic information for all local and federated groups for this tenant account.

   If the tenant account has the **Use grid federation connection** permission and you are viewing groups on the tenant's source grid:

   - A banner message indicates that if you edit or remove a group, your changes will not be synced to the other grid.
   - As needed, a banner message indicates if groups were not cloned to the tenant on the destination grid. You can retry a group clone that failed.
3. If you want to change the group's name:
   a. Select the checkbox for the group.
   b. Select **Actions** > **Edit group name**.
   c. Enter the new name.
   d. Select **Save changes.**
4. If you want to view more details or make additional edits, do either of the following:
   - Select the group name.
   - Select the checkbox for the group, and select **Actions** > **View group details**.
5. Review the Overview section, which shows the following information for each group:
   - Display name
   - Unique name
   - Type
   - Access mode
   - Permissions

- S3 Policy
- Number of users in this group
- Additional fields if the tenant account has the **Use grid federation connection** permission and you are viewing the group on the tenant's source grid:
  - Cloning status, either **Success** or **Failure**
  - A blue banner indicating that if you edit or delete this group, your changes will not be synced to the other grid.

6. Edit group settings as needed. See Create groups for an S3 tenant and Create groups for a Swift tenant for details about what to enter.
   a. In the Overview section, change the display name by selecting the name or the edit icon .
   b. On the **Group permissions** tab, update the permissions, and select **Save changes**.
   c. On the **Group policy** tab, make any changes, and select **Save changes**.
      - If you are editing an S3 group, optionally select a different S3 group policy or enter the JSON string for a custom policy, as required.
      - If you are editing a Swift group, optionally select or clear the **Swift Administrator** checkbox.
7. To add one or more existing local users to the group:
   a. Select the Users tab.



   b. Select **Add users**.
   c. Select the existing users you want to add, and select **Add users**.

      A success message appears in the upper right.

8. To remove local users from the group:
   a. Select the Users tab.
   b. Select **Remove users**.
   c. Select the users you want to remove, and select **Remove users**.

      A success message appears in the upper right.

9. Confirm that you selected **Save changes** for each section you changed.

Duplicate group

You can duplicate an existing group to create new groups more quickly.

> If your tenant account has the **Use grid federation connection** permission and you duplicate a group from the tenant's source grid, the duplicated group will be cloned to the tenant's destination grid.

Steps

1. Select **ACCESS MANAGEMENT** > **Groups**.
2. Select the checkbox for the group you want to duplicate.
3. Select **Actions** > **Duplicate group**.
4. See Create groups for an S3 tenant or Create groups for a Swift tenant for details about what to enter.
5. Select **Create group**.

Retry group clone

To retry a clone that failed:

1. Select each group that indicates *(Cloning failed)* below the group name.
2. Select **Actions** > **Clone groups**.
3. View the status of the clone operation from the details page of each group you're cloning.

For additional information, see Clone tenant groups and users.

Delete one or more groups

You can delete one or more groups. Any users who belong only to a group that is deleted will no longer be able to sign in to the Tenant Manager or use the tenant account.

> If your tenant account has the **Use grid federation connection** permission and you delete a group, StorageGRID will not delete the corresponding group on the other grid. If you need to keep this information in sync, you must delete the same group from both grids.

Steps

1. Select **ACCESS MANAGEMENT** > **Groups**.
2. Select the checkbox for each group you want to delete.
3. Select **Actions** > **Delete group** or **Actions** > **Delete groups**.

   A confirmation dialog box appears.

4. Select **Delete group** or **Delete groups**.

Manage local users

You can create local users and assign them to local groups to determine which features these users can access. The Tenant Manager includes one predefined local user, named "root." Although you can add and remove local users, you can't remove the root user.

If single sign-on (SSO) is enabled for your StorageGRID system, local users will not be able to sign in to the Tenant Manager or the Tenant Management API, although they can use client applications to access the tenant's resources, based on group permissions.

Before you begin

- You are signed in to the Tenant Manager using a supported web browser.
- You belong to a user group that has the Root access permission.
- If your tenant account has the **Use grid federation connection** permission, you have reviewed the workflow and considerations for cloning tenant groups and users, and you are signed in to the tenant's source grid.

Create a local user

You can create a local user and assign them to one or more local groups to control their access permissions.

S3 users who don't belong to any groups don't have management permissions or S3 group policies applied to them. These users might have S3 bucket access granted through a bucket policy.

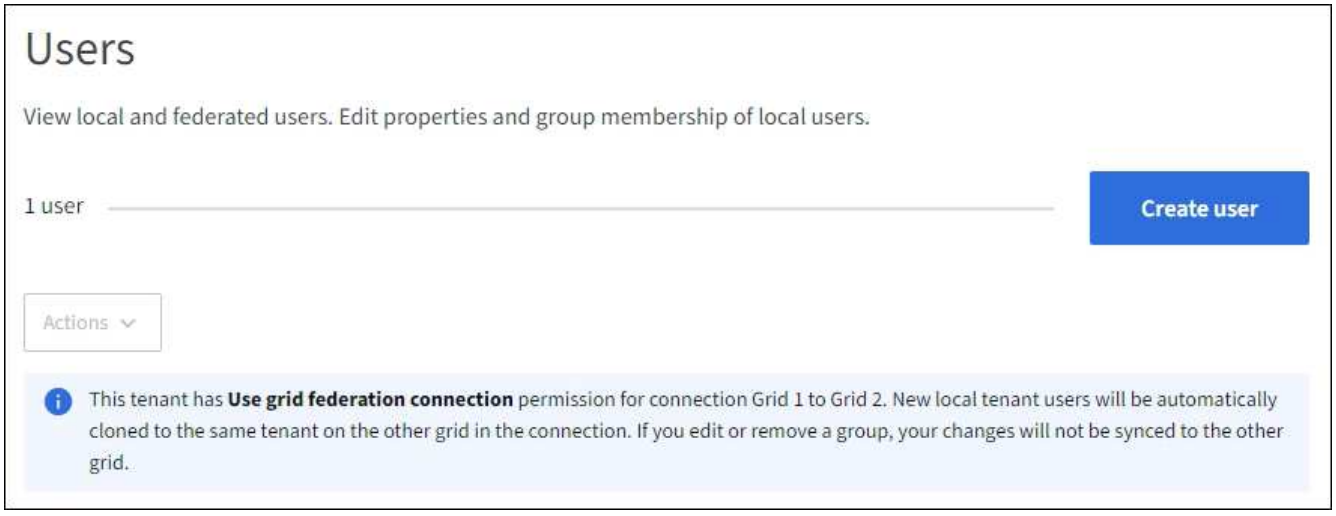
Swift users who don't belong to any groups don't have management permissions or Swift container access.

Access the Create user wizard

Steps

1. Select **ACCESS MANAGEMENT** > **Users**.

   If your tenant account has the **Use grid federation connection** permission, a blue banner indicates that this is the tenant's source grid. Any local users you create on this grid will be cloned to the other grid in the connection.



2. Select **Create user**.

Enter credentials

Steps

1. For the **Enter user credentials** step, complete the following fields.

Field	Description
Full name	The full name for this user, for example, the first name and last name of a person or the name of an application.
Username	The name this user will use to sign in. Usernames must be unique and can't be changed. **Note**: If your tenant account has the **Use grid federation connection** permission, a cloning error will occur if the same **Username** already exists for the tenant on the destination grid.
Password and Confirm password	The password the user will initially use when signing in.
Deny access	Select **Yes** to prevent this user from signing in to the tenant account, even though they might still belong to one or more groups. For example, select **Yes** to temporarily suspend a user's ability to sign in.

2. Select **Continue**.

Assign to groups

Steps

1. Assign the user to one or more local groups to determine which tasks they can perform.

   Assigning a user to groups is optional. If you'd prefer, you can select users when you create or edit groups.

   Users who don't belong to any groups will have no management permissions. Permissions are cumulative. Users will have all permissions for all groups they belong to. See Tenant management permissions.

2. Select **Create user**.

   If your tenant account has the **Use grid federation connection** permission and you are on the tenant's source grid, the new local user is cloned to the tenant's destination grid. **Success** appears as the **Cloning status** in the Overview section of the user's detail page.

3. Select **Finish** to return to the Users page.

View or edit local user

Steps

1. Select **ACCESS MANAGEMENT** > **Users**.
2. Review the information provided on the Users page, which lists basic information for all local and federated users for this tenant account.

   If the tenant account has the **Use grid federation connection** permission and you are viewing the user on the tenant's source grid:

   - A banner message indicates that if you edit or remove a user, your changes will not be synced to the other grid.

- As needed, a banner message indicates if users were not cloned to the tenant on the destination grid. You can retry a user clone that failed.

3. If you want to change the user's full name:
   a. Select the checkbox for the user.
   b. Select **Actions** > **Edit full name**.
   c. Enter the new name.
   d. Select **Save changes.**
4. If you want to view more details or make additional edits, do either of the following:
   - Select the username.
   - Select the checkbox for the user, and select **Actions** > **View user details**.
5. Review the Overview section, which shows the following information for each user:
   - Full name
   - Username
   - User type
   - Denied access
   - Access mode
   - Group membership
   - Additional fields if the tenant account has the **Use grid federation connection** permission and you are viewing the user on the tenant's source grid:
     - Cloning status, either **Success** or **Failure**
     - A blue banner indicating that if you edit this user, your changes will not be synced to the other grid.
6. Edit user settings as needed. See Create local user for details about what to enter.
   a. In the Overview section, change the full name by selecting the name or the edit icon .

      You can't change the username.

   b. On the **Password** tab, change the user's password, and select **Save changes**.
   c. On the **Access** tab, select **No** to allow the user to sign in or select **Yes** to prevent the user from signing in. Then, select **Save changes**.
   d. On the **Access keys** tab, select **Create key** and follow the instructions for creating another user's S3 access keys.
   e. On the **Groups** tab, select **Edit groups** to add the user to groups or remove the user from groups. Then, select **Save changes**.
7. Confirm that you selected **Save changes** for each section you changed.

Duplicate local user

You can duplicate a local user to create a new user more quickly.

> If your tenant account has the **Use grid federation connection** permission and you duplicate a user from the tenant's source grid, the duplicated user will be cloned to the tenant's destination grid.

Steps

1. Select **ACCESS MANAGEMENT** > **Users**.
2. Select the checkbox for the user you want to duplicate.
3. Select **Actions** > **Duplicate user**.
4. See Create local user for details about what to enter.
5. Select **Create user**.

Retry user clone

To retry a clone that failed:

1. Select each user that indicates *(Cloning failed)* below the user name.
2. Select **Actions** > **Clone users**.
3. View the status of the clone operation from the details page of each user you're cloning.

For additional information, see Clone tenant groups and users.

Delete one or more local users

You can permanently delete one or more local users who no longer need to access the StorageGRID tenant account.

> If your tenant account has the **Use grid federation connection** permission and you delete a local user, StorageGRID will not delete the corresponding user on the other grid. If you need to keep this information in sync, you must delete the same user from both grids.

> You must use the federated identity source to delete federated users.

Steps

1. Select **ACCESS MANAGEMENT** > **Users**.
2. Select the checkbox for each user you want to delete.
3. Select **Actions** > **Delete user** or **Actions** > **Delete users**.

   A confirmation dialog box appears.

4. Select **Delete user** or **Delete users**.

Manage S3 access keys

Manage S3 access keys

Each user of an S3 tenant account must have an access key to store and retrieve objects in the StorageGRID system. An access key consists of an access key ID and a secret access key.

S3 access keys can be managed as follows:

- Users who have the **Manage your own S3 credentials** permission can create or remove their own S3

access keys.

- Users who have the **Root access** permission can manage the access keys for the S3 root account and all other users. Root access keys provide full access to all buckets and objects for the tenant unless explicitly disabled by a bucket policy.

StorageGRID supports Signature Version 2 and Signature Version 4 authentication. Cross-account access is not permitted unless explicitly enabled by a bucket policy.

Create your own S3 access keys

If you are using an S3 tenant and you have the appropriate permission, you can create your own S3 access keys. You must have an access key to access your buckets and objects.

Before you begin

- You are signed in to the Tenant Manager using a supported web browser.
- You belong to a user group that has the Manage your own S3 credentials or Root access permission.

About this task

You can create one or more S3 access keys that allow you to create and manage buckets for your tenant account. After you create a new access key, update the application with your new access key ID and secret access key. For security, don't create more keys than you need, and delete the keys you aren't using. If you have only one key and it is about to expire, create a new key before the old one expires, and then delete the old one.

Each key can have a specific expiration time or no expiration. Follow these guidelines for expiration time:

- Set an expiration time for your keys to limit your access to a certain time period. Setting a short expiration time can help reduce your risk if your access key ID and secret access key are accidentally exposed. Expired keys are removed automatically.
- If the security risk in your environment is low and you don't need to periodically create new keys, you don't have to set an expiration time for your keys. If you decide later to create new keys, delete the old keys manually.

> The S3 buckets and objects belonging to your account can be accessed using the access key ID and secret access key displayed for your account in the Tenant Manager. For this reason, protect access keys as you would a password. Rotate access keys on a regular basis, remove any unused keys from your account, and never share them with other users.

Steps

1. Select **STORAGE (S3)** > **My access keys**.

   The My access keys page appears and lists any existing access keys.

2. Select **Create key**.
3. Do one of the following:
   - Select **Do not set an expiration time** to create a key that will not expire. (Default)
   - Select **Set an expiration time**, and set the expiration date and time.

The expiration date can be a maximum of five years from the current date. The expiration time can be a minimum of one minute from the current time.

4. Select **Create access key**.

   The Download access key dialog box appears, listing your access key ID and secret access key.

5. Copy the access key ID and the secret access key to a safe location, or select **Download .csv** to save a spreadsheet file containing the access key ID and secret access key.

   Don't close this dialog box until you have copied or downloaded this information. You can't copy or download keys after the dialog box has been closed.

6. Select **Finish**.

   The new key is listed on the My access keys page.

7. If your tenant account has the **Use grid federation connection** permission, optionally use the Tenant Management API to manually clone S3 access keys from the tenant on the source grid to the tenant on the destination grid. See Clone S3 access keys using the API.

View your S3 access keys

If you are using an S3 tenant and you have the appropriate permission, you can view a list of your S3 access keys. You can sort the list by expiration time, so you can determine which keys will expire soon. As needed, you can create new keys or delete keys that you are no longer using.

The S3 buckets and objects belonging to your account can be accessed using the access key ID and secret access key displayed for your account in the Tenant Manager. For this reason, protect access keys as you would a password. Rotate access keys on a regular basis, remove any unused keys from your account, and never share them with other users.

Before you begin

- You are signed in to the Tenant Manager using a supported web browser.
- You belong to a user group that has the Manage your own S3 credentials permission.

Steps

1. Select **STORAGE (S3)** > **My access keys**.
2. From the My access keys page, sort any existing access keys by **Expiration time** or **Access key ID**.
3. As needed, create new keys or delete any keys that you are no longer using.

   If you create new keys before the existing keys expire, you can begin using the new keys without temporarily losing access to the objects in the account.

   Expired keys are removed automatically.

Delete your own S3 access keys

If you are using an S3 tenant and you have the appropriate permission, you can delete your own S3 access keys. After an access key is deleted, it can no longer be used to access the objects and buckets in the tenant account.

Before you begin

- You are signed in to the Tenant Manager using a supported web browser.
- You have the Manage your own S3 credentials permission.

> The S3 buckets and objects belonging to your account can be accessed using the access key ID and secret access key displayed for your account in the Tenant Manager. For this reason, protect access keys as you would a password. Rotate access keys on a regular basis, remove any unused keys from your account, and never share them with other users.

Steps

1. Select **STORAGE (S3)** > **My access keys**.
2. From the My access keys page, select the checkbox for each access key you want to remove.
3. Select **Delete key**.
4. From the confirmation dialog box, select **Delete key**.

   A confirmation message appears in the upper right corner of the page.

Create another user's S3 access keys

If you are using an S3 tenant and you have the appropriate permission, you can create S3 access keys for other users, such as applications that need access to buckets and objects.

Before you begin

- You are signed in to the Tenant Manager using a supported web browser.
- You belong to a user group that has the Root access permission.

About this task

You can create one or more S3 access keys for other users so they can create and manage buckets for their tenant account. After you create a new access key, update the application with the new access key ID and secret access key. For security, don't create more keys than the user needs, and delete the keys that aren't being used. If you have only one key and it is about to expire, create a new key before the old one expires, and then delete the old one.

Each key can have a specific expiration time or no expiration. Follow these guidelines for expiration time:

- Set an expiration time for the keys to limit the user's access to a certain time period. Setting a short expiration time can help reduce risk if the access key ID and secret access key are accidentally exposed. Expired keys are removed automatically.
- If the security risk in your environment is low and you don't need to periodically create new keys, you don't have to set an expiration time for the keys. If you decide later to create new keys, delete the old keys manually.

The S3 buckets and objects belonging to a user can be accessed using the access key ID and secret access key displayed for that user in the Tenant Manager. For this reason, protect access keys as you would a password. Rotate access keys on a regular basis, remove any unused keys from the account, and never share them with other users.

Steps

1. Select **ACCESS MANAGEMENT** > **Users**.
2. Select the user whose S3 access keys you want to manage.

   The user detail page appears.

3. Select **Access keys**, then select **Create key**.
4. Do one of the following:
   - Select **Don't set an expiration time** to create a key that does not expire. (Default)
   - Select **Set an expiration time**, and set the expiration date and time.

     The expiration date can be a maximum of five years from the current date. The expiration time can be a minimum of one minute from the current time.

5. Select **Create access key**.

   The Download access key dialog box appears, listing the access key ID and secret access key.

6. Copy the access key ID and the secret access key to a safe location, or select **Download .csv** to save a spreadsheet file containing the access key ID and secret access key.

   Don't close this dialog box until you have copied or downloaded this information. You can't copy or download keys after the dialog box has been closed.

7. Select **Finish**.

   The new key is listed on the Access keys tab of the user details page.

8. If your tenant account has the **Use grid federation connection** permission, optionally use the Tenant Management API to manually clone S3 access keys from the tenant on the source grid to the tenant on the destination grid. See Clone S3 access keys using the API.

View another user's S3 access keys

If you are using an S3 tenant and you have appropriate permissions, you can view another user's S3 access keys. You can sort the list by expiration time so you can determine which keys will expire soon. As needed, you can create new keys and delete keys that are no longer in use.

Before you begin

- You are signed in to the Tenant Manager using a supported web browser.
- You have the Root access permission.

The S3 buckets and objects belonging to a user can be accessed using the access key ID and secret access key displayed for that user in the Tenant Manager. For this reason, protect access keys as you would a password. Rotate access keys on a regular basis, remove any unused keys from the account, and never share them with other users.

Steps

1. Select **ACCESS MANAGEMENT** > **Users**.
2. From the Users page, select the user whose S3 access keys you want to view.
3. From the User details page, select **Access keys**.
4. Sort the keys by **Expiration time** or **Access key ID**.
5. As needed, create new keys and manually delete keys that the are no longer in use.

   If you create new keys before the existing keys expire, the user can begin using the new keys without temporarily losing access to the objects in the account.

   Expired keys are removed automatically.

Related information

- Create another user's S3 access keys
- Delete another user's S3 access keys

Delete another user's S3 access keys

If you are using an S3 tenant and you have appropriate permissions, you can delete another user's S3 access keys. After an access key is deleted, it can no longer be used to access the objects and buckets in the tenant account.

Before you begin

- You are signed in to the Tenant Manager using a supported web browser.
- You have the Root access permission.

The S3 buckets and objects belonging to a user can be accessed using the access key ID and secret access key displayed for that user in the Tenant Manager. For this reason, protect access keys as you would a password. Rotate access keys on a regular basis, remove any unused keys from the account, and never share them with other users.

Steps

1. Select **ACCESS MANAGEMENT** > **Users**.
2. From the Users page, select the user whose S3 access keys you want to manage.
3. From the User details page, select **Access keys**, and then select the checkbox for each access key you want to delete.
4. Select **Actions** > **Delete selected key**.
5. From the confirmation dialog box, select **Delete key**.

   A confirmation message appears in the upper right corner of the page.

Manage S3 buckets

Create an S3 bucket

You can use the Tenant Manager to create S3 buckets for object data.

Before you begin

- You are signed in to the Tenant Manager using a supported web browser.
- You belong to a user group that has the Root access or Manage all buckets permission. These permissions override the permissions settings in group or bucket policies.

> Permissions to set or modify S3 Object Lock properties of buckets or objects can be granted by bucket policy or group policy.

- If you plan to enable S3 Object Lock for a bucket, a grid admin has enabled the global S3 Object Lock setting for the StorageGRID system, and you have reviewed the requirements for S3 Object Lock buckets and objects.
- If each tenant will have 5,000 buckets, each Storage Node in the grid has a minimum of 64 GB of RAM.

> Each grid can have a maximum of 100,000 buckets.

Access the wizard

Steps

1. Select **View buckets** from the dashboard, or select **STORAGE (S3)** > **Buckets**.
2. Select **Create bucket**.

Enter details

Steps

1. Enter details for the bucket.

Field	Description
Bucket name	A name for the bucket that complies with these rules: • Must be unique across each StorageGRID system (not just unique within the tenant account). • Must be DNS compliant. • Must contain at least 3 and no more than 63 characters. • Each label must start and end with a lowercase letter or a number and can only use lowercase letters, numbers, and hyphens. • Must not contain periods in virtual hosted style requests. Periods will cause problems with server wildcard certificate verification. For more information, see the Amazon Web Services (AWS) documentation on bucket naming rules. **Note**: You can't change the bucket name after creating the bucket.
Region	The bucket's region. Your StorageGRID administrator manages the available regions. A bucket's region can affect the data-protection policy applied to objects. By default, all buckets are created in the `us-east-1` region. **Note**: You can't change the region after creating the bucket.

2. Select **Continue**.

Manage settings

Steps

1. Optionally, enable object versioning for the bucket.

   Enable object versioning if you want to store every version of each object in this bucket. You can then retrieve previous versions of an object as needed. You must enable object versioning if the bucket will be used for cross-grid replication.

2. If the global S3 Object Lock setting is enabled, optionally enable S3 Object Lock for the bucket to store objects using a write-once-read-many (WORM) model.

   Enable S3 Object Lock for a bucket only if you need to keep objects for fixed amount of time, for example, to meet certain regulatory requirements. S3 Object Lock is a permanent setting that helps you prevent objects from being deleted or overwritten for a fixed amount of time or indefinitely.

   > After the S3 Object Lock setting is enabled for a bucket, it can't be disabled. Anyone with the correct permissions can add objects to this bucket that can't be changed. You might not be able to delete these objects or the bucket itself.

   If you enable S3 Object Lock for a bucket, bucket versioning is enabled automatically.

3. If you selected **Enable S3 Object Lock**, optionally enable **Default retention** for this bucket.

Your grid administrator must give you permission to use specific features of S3 Object Lock.

When **Default retention** is enabled, new objects added to the bucket will be automatically protected from being deleted or overwritten. The **Default retention** setting does not apply to objects that have their own retention periods.

a. If **Default retention** is enabled, specify a **Default retention mode** for the bucket.

Default retention mode	Description
Governance	• Users with the `s3:BypassGovernanceRetention` permission can use the `x-amz-bypass-governance-retention: true` request header to bypass retention settings. • These users can delete an object version before its retain-until-date is reached. • These users can increase, decrease, or remove an object's retain-until-date.
Compliance	• The object can't be deleted until its retain-until-date is reached. • The object's retain-until-date can be increased, but it can't be decreased. • The object's retain-until-date can't be removed until that date is reached. **Note**: Your grid administrator must allow you to use compliance mode.

b. If **Default retention** is enabled, specify the **Default retention period** for the bucket.

The **Default retention period** indicates how long new objects added to this bucket should be retained, starting from the time they are ingested. Specify a value that is less than or equal to the maximum retention period for the tenant, as set by the grid administrator.

A *maximum* retention period, which can be a value from 1 day to 100 years, is set when the grid administrator creates the tenant. When you set a *default* retention period, it can't exceed the value set for the maximum retention period. If needed, ask your grid administrator to increase or decrease the maximum retention period.

4. Optionally, select **Enable capacity limit**.

Capacity limit is the maximum capacity available for this bucket's objects. This value represents a logical amount (object size), not a physical amount (size on disk).

If no limit is set, the capacity for this bucket is unlimited. Refer to Capacity limit usage for more information.

5. Select **Create bucket**.

The bucket is created and added to the table on the Buckets page.

6. Optionally, select **Go to bucket details page** to view bucket details and perform additional configuration.

View bucket details

You can view the buckets in your tenant account.

Before you begin

- You are signed in to the Tenant Manager using a supported web browser.
- You belong to a user group that has the Root access, Manage all buckets, or View all buckets permission. These permissions override the permission settings in group or bucket policies.

Steps

1. Select **View buckets** from the dashboard, or select **STORAGE (S3)** > **Buckets**.

   The Buckets page appears.

2. Review the summary table for each bucket.

   As required, you can sort the information by any column, or you can page forward and back through the list.

   > The Object Count, Space Used, and Usage values displayed are estimates. These estimates are affected by the timing of ingests, network connectivity, and node status. If buckets have versioning enabled, deleted object versions are included in the object count.

   Name
   The bucket's unique name, which can't be changed.

   Enabled features
   The list of features that are enabled for the bucket.

   S3 Object Lock
   Whether S3 Object Lock is enabled for the bucket.

   This column appears only if S3 Object Lock is enabled for the grid. This column also shows information for any legacy Compliant buckets.

   Region
   The bucket's region, which can't be changed. This column is hidden by default.

   Object count
   The number of objects in this bucket. If buckets have versioning enabled, non-current object versions are included in this value.

   When objects are added or deleted, this value might not update immediately.

   Space used
   The logical size of all objects in the bucket. The logical size does not include the actual space required for replicated or erasure-coded copies or for object metadata.

   This value can take up to 10 minutes to update.

Usage

The percentage used of the bucket's capacity limit, if one has been set.

The usage value is based on internal estimates and might be exceeded in some cases. For example, StorageGRID checks capacity limit (if set) when a tenant starts uploading objects and rejects new ingests to this bucket if the tenant has exceeded the capacity limit. However, StorageGRID does not take into account the size of the current upload when determining if the capacity limit has been exceeded. If objects are deleted, a tenant might be temporarily prevented from uploading new objects to this bucket until the capacity limit usage is recalculated. The calculations can take 10 minutes or longer.

This value indicates logical size, not physical size needed to store the objects and their metadata.

Capacity

If set, the capacity limit for the bucket.

Date created

The date and time the bucket was created. This column is hidden by default.

3. To view details for a specific bucket, select the bucket name from the table.
   a. View the summary information at the top of the web page to confirm the details for the bucket, such as Region and Object count.
   b. View the Capacity limit usage bar. If the usage is 100% or near 100%, consider increasing the limit or deleting some objects.
   c. As needed, select **Delete objects in bucket** and **Delete bucket**.

      Pay close attention to the cautions that appear when you select each of these options. For more information, refer to:

      - Delete all objects in a bucket
      - Delete a bucket (bucket must be empty)

   d. View or change settings for the bucket in each of the tabs as needed.
      - **S3 Console**: View the objects for the bucket. For more information, refer to Use S3 Console.
      - **Bucket options**: View or change option settings. Some settings, such as S3 Object Lock, can't be changed after the bucket is created.
        - Manage bucket consistency
        - Last access time updates
        - Capacity limit
        - Object versioning
        - S3 Object Lock
        - Default bucket retention
        - Manage cross-grid replication (if allowed for the tenant)
      - **Platform services**: Manage platform services (if allowed for the tenant)
      - **Bucket access**: View or change option settings. You must have specific access permissions.
        - Configure Cross-Origin Resource Sharing (CORS) so the bucket and objects in the bucket will be accessible to web applications in other domains.

- Control user access for an S3 bucket and objects in that bucket.

Apply an ILM policy tag to a bucket

Choose an ILM policy tag to apply to a bucket based on your object storage requirements.

The ILM policy controls where the object data is stored and whether it is deleted after a certain time period. Your grid administrator creates ILM policies and assigns them to ILM policy tags when using multiple active policies.

Avoid frequently reassigning a bucket's policy tag. Otherwise, performance issues might occur.

Before you begin

- You are signed in to the Tenant Manager using a supported web browser.
- You belong to a user group that has the Root access, Manage all buckets, or View all buckets permission. These permissions override the permission settings in group or bucket policies.

Steps

1. Select **View buckets** from the dashboard, or select **STORAGE (S3)** > **Buckets**.

   The Buckets page appears. As required, you can sort the information by any column, or you can page forward and back through the list.

2. Select the name of the bucket you want to assign an ILM policy tag to.

   You can also change the ILM policy tag assignment for a bucket that already has a tag assigned.

   The Object Count and Space Used values displayed are estimates. These estimates are affected by the timing of ingests, network connectivity, and node status. If buckets have versioning enabled, deleted object versions are included in the object count.

3. In the Bucket options tab, expand the ILM policy tag accordion. This accordion only appears if your grid administrator has enabled the use of custom policy tags.
4. Read the description of each policy tag to determine which tag should be applied to the bucket.

   Changing the ILM policy tag for a bucket will trigger ILM reevaluation of all objects in the bucket. If the new policy retains objects for a limited time, older objects will be deleted.

5. Select the radio button for the tag you want to assign to the bucket.
6. Select **Save changes**. A new S3 bucket tag will be set on the bucket with the key `NTAP-SG-ILM-BUCKET-TAG` and the value of the ILM policy tag name.

   Ensure that your S3 applications do not accidentally override or delete the new bucket tag. If this tag is omitted when applying a new TagSet to the bucket, objects in the bucket will revert to being evaluated against the default ILM policy.

Set and modify ILM policy tags using only the Tenant Manager or Tenant Manager API where the ILM policy tag is validated. Do not modify the `NTAP-SG-ILM-BUCKET-TAG` ILM policy tag using the S3 PutBucketTagging API or the S3 DeleteBucketTagging API.

Changing the policy tag assigned to a bucket has a temporary performance impact while objects are being reevaluated using the new ILM policy.

Manage bucket policy

You can control user access for an S3 bucket and the objects in that bucket.

Before you begin

- You are signed in to the Tenant Manager using a supported web browser.
- You belong to a user group that has the Root access permission. The View all buckets and Manage all buckets permissions only allow viewing.
- You've verified that the required number of Storage Nodes and sites are available. If two or more Storage Nodes are not available within any site, or if a site is not available, changes to these settings might not be available.

Steps

1. Select **Buckets**, then select the bucket you want to manage.
2. On the bucket details page, select **Bucket access** > **Bucket policy**.
3. Do one of the following:
   - Enter a bucket policy by selecting the **Enable policy** checkbox. Then enter a valid JSON formatted string.

     Each bucket policy has a size limit of 20,480 bytes.
   - Modify an existing policy by editing the string.
   - Disable a policy by unselecting **Enable policy**.

For detailed information about bucket policies, including language syntax and examples, see Example bucket policies.

Manage bucket consistency

Consistency values can be used to specify the availability of bucket setting changes as well as to provide a balance between the availability of the objects within a bucket and the consistency of those objects across different Storage Nodes and sites. You can change the consistency values to be different from the default values so that client applications can meet their operational needs.

Before you begin

- You are signed in to the Tenant Manager using a supported web browser.
- You belong to a user group that has the Manage all buckets or Root access permission. These permissions override the permissions settings in group or bucket policies.

Bucket consistency guidelines

The bucket consistency is used to determine the consistency for client applications affecting objects within that S3 bucket. In general, you should use the **Read-after-new-write** consistency for your buckets.

Change bucket consistency

If the **Read-after-new-write** consistency does not meet the client application's requirements, you can change the consistency by setting the bucket consistency or by using the `Consistency-Control` header. The `Consistency-Control` header overrides the bucket consistency.

> When you change a bucket's consistency, only those objects that are ingested after the change are guaranteed to meet the revised setting.

Steps

1. Select **View buckets** from the dashboard, or select **STORAGE (S3)** > **Buckets**.
2. Select the bucket name from the table.

   The bucket details page appears.
3. From the **Bucket options** tab, select the ** accordion.
4. Select a consistency for operations performed on the objects in this bucket.
   - **All**: Provides the highest level of consistency. All nodes receive the data immediately, or the request will fail.
   - **Strong-global**: Guarantees read-after-write consistency for all client requests across all sites.
   - **Strong-site**: Guarantees read-after-write consistency for all client requests within a site.
   - **Read-after-new-write** (default): Provides read-after-write consistency for new objects and eventual consistency for object updates. Offers high availability and data protection guarantees. Recommended for most cases.
   - **Available**: Provides eventual consistency for both new objects and object updates. For S3 buckets, use only as required (for example, for a bucket that contains log values that are rarely read, or for HEAD or GET operations on keys that don't exist). Not supported for S3 FabricPool buckets.
5. Select **Save changes**.

What happens when you change bucket settings

Buckets have multiple settings that affect the behavior of the buckets and the objects within those buckets.

The following bucket settings use **strong** consistency by default. If two or more Storage Nodes are not available within any site, or if a site is not available, any changes to these settings might not be available.

- Background empty bucket deletion
- Last Access Time
- Bucket lifecycle
- Bucket policy
- Bucket tagging
- Bucket versioning

- S3 Object Lock
- Bucket encryption

> The consistency value for bucket versioning, S3 Object Lock, and bucket encryption cannot be set to a value that is not strongly consistent.

The following bucket settings do not use strong consistency and have higher availability for changes. Changes to these settings might take some time before having an effect.

- Platform services configuration: Notification, Replication, or Search integration
- CORS configuration
- Change bucket consistency

> If the default consistency used when changing bucket settings does not meet the client application's requirements, you can change the consistency by using the `Consistency-Control` header for the S3 REST API or by using the `reducedConsistency` or `force` options in the Tenant Management API.

Enable or disable last access time updates

When grid administrators create the information lifecycle management (ILM) rules for a StorageGRID system, they can optionally specify that an object's last access time be used to determine whether to move that object to a different storage location. If you are using an S3 tenant, you can take advantage of such rules by enabling last access time updates for the objects in an S3 bucket.

These instructions only apply to StorageGRID systems that include at least one ILM rule that uses the **Last access time** option as an advanced filter or as a reference time. You can ignore these instructions if your StorageGRID system does not include such a rule. See Use Last access time in ILM rules for details.

Before you begin

- You are signed in to the Tenant Manager using a supported web browser.
- You belong to a user group that has the Manage all buckets or Root access permission. These permissions override the permissions settings in group or bucket policies.

About this task

Last access time is one of the options available for the **Reference time** placement instruction for an ILM rule. Setting the Reference time for a rule to Last access time lets grid administrators specify that objects be placed in certain storage locations based on when those objects were last retrieved (read or viewed).

For example, to ensure that recently viewed objects remain on faster storage, a grid administrator can create an ILM rule specifying the following:

- Objects that have been retrieved in the past month should remain on local Storage Nodes.
- Objects that have not been retrieved in the past month should be moved to an off-site location.

By default, updates to last access time are disabled. If your StorageGRID system includes an ILM rule that uses the **Last access time** option and you want this option to apply to objects in this bucket, you must enable updates to last access time for the S3 buckets specified in that rule.

Updating the last access time when an object is retrieved can reduce StorageGRID performance, especially for small objects.

A performance impact occurs with last access time updates because StorageGRID must perform these additional steps every time objects are retrieved:

- Update the objects with new timestamps
- Add the objects to the ILM queue, so they can be reevaluated against current ILM rules and policy

The table summarizes the behavior applied to all objects in the bucket when last access time is disabled or enabled.

Type of request	Behavior if last access time is disabled (default)		Behavior if last access time is enabled	
	Last access time updated?	Object added to ILM evaluation queue?	Last access time updated?	Object added to ILM evaluation queue?
Request to retrieve an object, its access control list, or its metadata	No	No	Yes	Yes
Request to update an object's metadata	Yes	Yes	Yes	Yes
Request to list objects or object versions	No	No	No	No
Request to copy an object from one bucket to another	• No, for the source copy • Yes, for the destination copy	• No, for the source copy • Yes, for the destination copy	• Yes, for the source copy • Yes, for the destination copy	• Yes, for the source copy • Yes, for the destination copy
Request to complete a multipart upload	Yes, for the assembled object	Yes, for the assembled object	Yes, for the assembled object	Yes, for the assembled object

Steps

1. Select **View buckets** from the dashboard, or select **STORAGE (S3)** > **Buckets**.
2. Select the bucket name from the table.

   The bucket details page appears.

3. From the **Bucket options** tab, select the **Last access time updates** accordion.
4. Enable or disable last access time updates.
5. Select **Save changes**.

Change object versioning for a bucket

If you are using an S3 tenant, you can change the versioning state for S3 buckets.

Before you begin

- You are signed in to the Tenant Manager using a supported web browser.
- You belong to a user group that has the Manage all buckets or Root access permission. These permissions override the permissions settings in group or bucket policies.
- You've verified that the required number of Storage Nodes and sites are available. If two or more Storage Nodes are not available within any site, or if a site is not available, changes to these settings might not be available.

About this task

You can enable or suspend object versioning for a bucket. After you enable versioning for a bucket, it can't return to an unversioned state. However, you can suspend versioning for the bucket.

- Disabled: Versioning has never been enabled
- Enabled: Versioning is enabled
- Suspended: Versioning was previously enabled and is suspended

For more information, see the following:

- Object versioning
- ILM rules and policies for S3 versioned objects (Example 4)
- How objects are deleted

Steps

1. Select **View buckets** from the dashboard, or select **STORAGE (S3)** > **Buckets**.
2. Select the bucket name from the table.

   The bucket details page appears.

3. From the **Bucket options** tab, select the **Object versioning** accordion.
4. Select a versioning state for the objects in this bucket.

   Object versioning must remain enabled for a bucket used for cross-grid replication. If S3 Object Lock or legacy compliance is enabled, the **Object versioning** options are disabled.

Option	Description
Enable versioning	Enable object versioning if you want to store every version of each object in this bucket. You can then retrieve previous versions of an object as needed. Objects that were already in the bucket will be versioned when they are modified by a user.
Suspend versioning	Suspend object versioning if you no longer want new object versions to be created. You can still retrieve any existing object versions.

5. Select **Save changes**.

Use S3 Object Lock to retain objects

You can use S3 Object Lock if buckets and objects must comply with regulatory requirements for retention.

Your grid administrator must give you permission to use specific features of S3 Object Lock.

What is S3 Object Lock?

The StorageGRID S3 Object Lock feature is an object-protection solution that is equivalent to S3 Object Lock in Amazon Simple Storage Service (Amazon S3).

When the global S3 Object Lock setting is enabled for a StorageGRID system, an S3 tenant account can create buckets with or without S3 Object Lock enabled. If a bucket has S3 Object Lock enabled, bucket versioning is required and is enabled automatically.

A bucket without S3 Object Lock can only have objects without retention settings specified. No ingested objects will have retention settings.

A bucket with S3 Object Lock can have objects with and without retention settings specified by S3 client applications. Some objects ingested will have retention settings.

A bucket with S3 Object Lock and default retention configured can have uploaded objects with retention settings specified and new objects without retention settings. The new objects use the default setting, because the retention setting hasn't been configured at the object-level.

Effectively, all newly ingested objects have retention settings when default retention is configured. Existing objects without object retention settings remain unaffected.

Retention modes

The StorageGRID S3 Object Lock feature supports two retention modes to apply different levels of protection to objects. These modes are equivalent to the Amazon S3 retention modes.

- In compliance mode:
  - The object can't be deleted until its retain-until-date is reached.
  - The object's retain-until-date can be increased, but it can't be decreased.
  - The object's retain-until-date can't be removed until that date is reached.
- In governance mode:
  - Users with special permission can use a bypass header in requests to modify certain retention settings.
  - These users can delete an object version before its retain-until-date is reached.
  - These users can increase, decrease, or remove an object's retain-until-date.

Retention settings for object versions

If a bucket is created with S3 Object Lock enabled, users can use the S3 client application to optionally specify the following retention settings for each object that is added to the bucket:

- **Retention mode**: Either compliance or governance.
- **Retain-until-date**: If an object version's retain-until-date is in the future, the object can be retrieved, but it can't be deleted.
- **Legal hold**: Applying a legal hold to an object version immediately locks that object. For example, you might need to put a legal hold on an object that is related to an investigation or legal dispute. A legal hold has no expiration date, but remains in place until it is explicitly removed. Legal holds are independent of the retain-until-date.

  > If an object is under a legal hold, no one can delete the object, regardless of its retention mode.

  For details on the object settings, see Use S3 REST API to configure S3 Object Lock.

Default retention setting for buckets

If a bucket is created with S3 Object Lock enabled, users can optionally specify the following default settings for the bucket:

- **Default retention mode**: Either compliance or governance.
- **Default retention period**: How long new object versions added to this bucket should be retained, starting from the day they are added.

The default bucket settings apply only to new objects that don't have their own retention settings. Existing bucket objects aren't affected when you add or change these default settings.

See Create an S3 bucket and Update S3 Object Lock default retention.

S3 Object Lock tasks

The following lists for grid administrators and tenant users contain the high-level tasks for using the S3 Object Lock feature.

Grid administrator

- Enable global S3 Object Lock setting for entire StorageGRID system.
- Ensure that information lifecycle management (ILM) policies are *compliant*; that is, they meet the requirements of buckets with S3 Object Lock enabled.
- As needed, allow a tenant to use Compliance as the retention mode. Otherwise, only Governance mode is allowed.
- As needed, set a maximum retention period for a tenant.

Tenant user

- Review considerations for buckets and objects with S3 Object Lock.
- As needed, contact grid administrator to enable global S3 Object Lock setting and set permissions.
- Create buckets with S3 Object Lock enabled.
- Optionally, configure default retention settings for a bucket:
  - Default retention mode: Governance or Compliance, if allowed by the grid administrator.
  - Default retention period: Must be less than or equal to maximum retention period set by grid administrator.

- Use the S3 client application to add objects and optionally set object-specific retention:
  - Retention mode. Governance or Compliance, if allowed by the grid administrator.
  - Retain Until Date: Must be less than or equal to what is allowed by the maximum retention period set by grid administrator.

Requirements for buckets with S3 Object Lock enabled

- If the global S3 Object Lock setting is enabled for the StorageGRID system, you can use the Tenant Manager, the Tenant Management API, or the S3 REST API to create buckets with S3 Object Lock enabled.
- If you plan to use S3 Object Lock, you must enable S3 Object Lock when you create the bucket. You can't enable S3 Object Lock for an existing bucket.
- When S3 Object Lock is enabled for a bucket, StorageGRID automatically enables versioning for that bucket. You can't disable S3 Object Lock or suspend versioning for the bucket.
- Optionally, you can specify a default retention mode and retention period for each bucket using the Tenant Manager, the Tenant Management API, or the S3 REST API. The bucket's default retention settings apply only to new objects added to the bucket that don't have their own retention settings. You can override these default settings by specifying a retention mode and retain-until-date for each object version when it is uploaded.
- Bucket lifecycle configuration is supported for buckets with S3 Object Lock enabled.
- CloudMirror replication is not supported for buckets with S3 Object Lock enabled.

Requirements for objects in buckets with S3 Object Lock enabled

- To protect an object version, you can specify default retention settings for the bucket, or you can specify retention settings for each object version. Object-level retention settings can be specified using the S3 client application or the S3 REST API.
- Retention settings apply to individual object versions. An object version can have both a retain-until-date and a legal hold setting, one but not the other, or neither. Specifying a retain-until-date or a legal hold setting for an object protects only the version specified in the request. You can create new versions of the object, while the previous version of the object remains locked.

Lifecycle of objects in buckets with S3 Object Lock enabled

Each object that is saved in a bucket with S3 Object Lock enabled goes through these stages:

1. **Object ingest**

   When an object version is added to bucket that has S3 Object Lock enabled, retention settings are applied as follows:

   - If retention settings are specified for the object, the object-level settings are applied. Any default bucket settings are ignored.
   - If no retention settings are specified for the object, the default bucket settings are applied, if they exist.
   - If no retention settings are specified for the object or the bucket, the object is not protected by S3 Object Lock.

   If retention settings are applied, both the object and any S3 user-defined metadata are protected.

2. **Object retention and deletion**

Multiple copies of each protected object are stored by StorageGRID for the specified retention period. The exact number and type of object copies and the storage locations are determined by the compliant rules in the active ILM policies. Whether a protected object can be deleted before its retain-until-date is reached depends on its retention mode.

- If an object is under a legal hold, no one can delete the object, regardless of its retention mode.

Can I still manage legacy Compliant buckets?

The S3 Object Lock feature replaces the Compliance feature that was available in previous StorageGRID versions. If you created compliant buckets using a previous version of StorageGRID, you can continue to manage the settings of these buckets; however, you can no longer create new compliant buckets. For instructions, see NetApp Knowledge Base: How to manage legacy Compliant buckets in StorageGRID 11.5.

Update S3 Object Lock default retention

If you enabled S3 Object Lock when you created the bucket, you can edit the bucket to change the default retention settings. You can enable (or disable) default retention and set a default retention mode and retention period.

Before you begin

- You are signed in to the Tenant Manager using a supported web browser.
- You belong to a user group that has the Manage all buckets or Root access permission. These permissions override the permissions settings in group or bucket policies.
- S3 Object Lock is enabled globally for your StorageGRID system, and you enabled S3 Object Lock when you created the bucket. See Use S3 Object Lock to retain objects.

Steps

1. Select **View buckets** from the dashboard, or select **STORAGE (S3)** > **Buckets**.
2. Select the bucket name from the table.

   The bucket details page appears.

3. From the **Bucket options** tab, select the **S3 Object Lock** accordion.
4. Optionally, enable or disable **Default retention** for this bucket.

   Changes to this setting don't apply to objects already in the bucket or to any objects that might have their own retention periods.

5. If **Default retention** is enabled, specify a **Default retention mode** for the bucket.

Default retention mode	Description
Governance	• Users with the `s3:BypassGovernanceRetention` permission can use the `x-amz-bypass-governance-retention: true` request header to bypass retention settings. • These users can delete an object version before its retain-until-date is reached. • These users can increase, decrease, or remove an object's retain-until-date.
Compliance	• The object can't be deleted until its retain-until-date is reached. • The object's retain-until-date can be increased, but it can't be decreased. • The object's retain-until-date can't be removed until that date is reached. **Note**: Your grid administrator must allow you to use compliance mode.

6. If **Default retention** is enabled, specify the **Default retention period** for the bucket.

   The **Default retention period** indicates how long new objects added to this bucket should be retained, starting from the time they are ingested. Specify a value that is less than or equal to the maximum retention period for the tenant, as set by the grid administrator.

   A *maximum* retention period, which can be a value from 1 day to 100 years, is set when the grid administrator creates the tenant. When you set a *default* retention period, it can't exceed the value set for the maximum retention period. If needed, ask your grid administrator to increase or decrease the maximum retention period.

7. Select **Save changes**.

Configure cross-origin resource sharing (CORS)

You can configure cross-origin resource sharing (CORS) for an S3 bucket if you want that bucket and objects in that bucket to be accessible to web applications in other domains.

Before you begin

- You are signed in to the Tenant Manager using a supported web browser.
- For GET CORS configuration requests, you belong to a user group that has the Manage all buckets or View all buckets permission. These permissions override the permissions settings in group or bucket policies.
- For PUT CORS configuration requests, you belong to a user group that has the Manage all buckets permission. This permission overrides the permissions settings in group or bucket policies.
- The Root access permission provides access to all CORS configuration requests.

About this task

Cross-origin resource sharing (CORS) is a security mechanism that allows client web applications in one domain to access resources in a different domain. For example, suppose you use an S3 bucket named

`Images` to store graphics. By configuring CORS for the `Images` bucket, you can allow the images in that bucket to be displayed on the website `http://www.example.com`.

Enable CORS for a bucket

Steps

1. Use a text editor to create the required XML. This example shows the XML used to enable CORS for an S3 bucket. Specifically:
   - Allows any domain to send GET requests to the bucket
   - Only allows the `http://www.example.com` domain to send GET, POST, and DELETE requests
   - All request headers are allowed

   ```
   <CORSConfiguration
       xmlns="http://s3.amazonaws.com/doc/2020-10-22/">
       <CORSRule>
           <AllowedOrigin>*</AllowedOrigin>
           <AllowedMethod>GET</AllowedMethod>
           <AllowedHeader>*</AllowedHeader>
       </CORSRule>
       <CORSRule>
           <AllowedOrigin>http://www.example.com</AllowedOrigin>
           <AllowedMethod>GET</AllowedMethod>
           <AllowedMethod>POST</AllowedMethod>
           <AllowedMethod>DELETE</AllowedMethod>
           <AllowedHeader>*</AllowedHeader>
       </CORSRule>
   </CORSConfiguration>
   ```

   For more information about the CORS configuration XML, see Amazon Web Services (AWS) Documentation: Amazon Simple Storage Service User Guide.

2. Select **View buckets** from the dashboard, or select **STORAGE (S3)** > **Buckets**.
3. Select the bucket name from the table.

   The bucket details page appears.

4. From the **Bucket access** tab, select the **Cross-Origin Resource Sharing (CORS)** accordion.
5. Select the **Enable CORS** checkbox.
6. Paste the CORS configuration XML into the text box.
7. Select **Save changes**.

Modify CORS setting

Steps

1. Update the CORS configuration XML in the text box, or select **Clear** to start over.
2. Select **Save changes**.

Disable CORS setting

Steps

1. Clear the **Enable CORS** checkbox.
2. Select **Save changes**.

Delete objects in bucket

You can use the Tenant Manager to delete the objects in one or more buckets.

Considerations and requirements

Before performing these steps, note the following:

- When you delete the objects in a bucket, StorageGRID permanently removes all objects and all object versions in each selected bucket from all nodes and sites in your StorageGRID system. StorageGRID also removes any related object metadata. You will not be able to recover this information.
- Deleting all of the objects in a bucket might take minutes, days, or even weeks, based on the number of objects, object copies, and concurrent operations.
- If a bucket has S3 Object Lock enabled, it might remain in the **Deleting objects: read-only** state for *years*.

> A bucket that uses S3 Object Lock will remain in the **Deleting objects: read-only** state until the retention date is reached for all objects and any legal holds are removed.

- While objects are being deleted, the bucket's state is **Deleting objects: read-only**. In this state, you can't add new objects to the bucket.
- When all objects have been deleted, the bucket remains in the read-only state. You can do one of the following:
  - Return the bucket to write mode and reuse it for new objects
  - Delete the bucket
  - Keep the bucket in read-only mode to reserve its name for future use
- If a bucket has object versioning enabled, delete markers that were created in StorageGRID 11.8 or later can be removed using the Delete objects in bucket operations.
- If a bucket has object versioning enabled, the delete objects operation will not remove delete markers that were created in StorageGRID 11.7 or earlier. See information about deleting objects in a bucket in How S3 versioned objects are deleted.
- If you use cross-grid replication, note the following:
  - Using this option does not delete any objects from the bucket on the other grid.
  - If you select this option for the source bucket, the **Cross-grid replication failure** alert will be triggered if you add objects to the destination bucket on the other grid. If you can't guarantee no one will add objects to the bucket on the other grid, disable cross-grid replication for that bucket before deleting all bucket objects.

Before you begin

- You are signed in to the Tenant Manager using a supported web browser.
- You belong to a user group that has the Root access permission. This permission overrides the permissions settings in group or bucket policies.

Steps

1. Select **View buckets** from the dashboard, or select **STORAGE (S3)** > **Buckets**.

   The Buckets page appears and shows all existing S3 buckets.

2. Use the **Actions** menu or the details page for a specific bucket.

   Actions menu

   a. Select the checkbox for each bucket you want to delete objects from.

   b. Select **Actions** > **Delete objects in bucket**.

   Details page

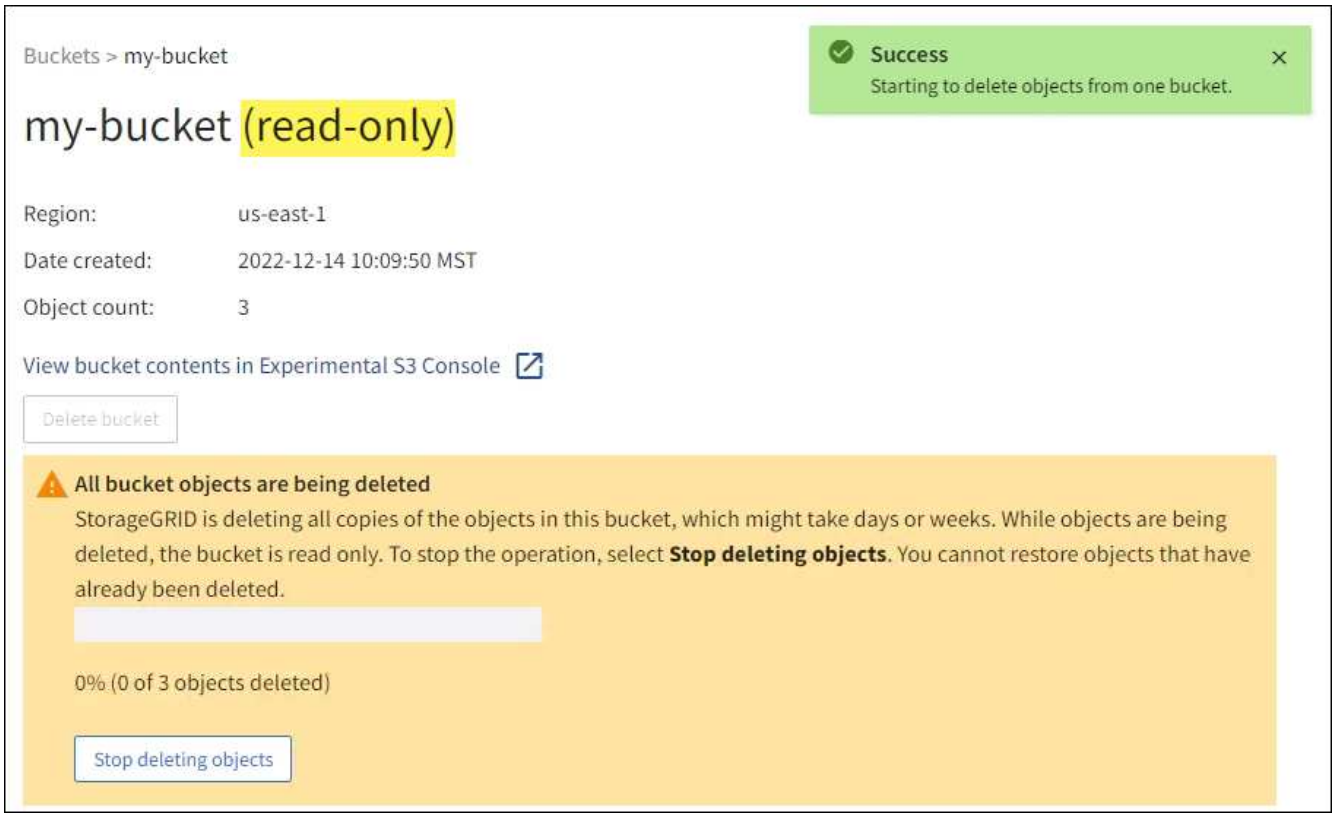
   a. Select a bucket name to display its details.

   b. Select **Delete objects in bucket**.

3. When the confirmation dialog box appears, review the details, enter **Yes**, and select **OK**.
4. Wait for the delete operation to begin.

   After a few minutes:

   - A yellow status banner appears on the bucket details page. The progress bar represents what percentage of objects have been deleted.
   - **(read-only)** appears after the bucket's name on the bucket details page.
   - **(Deleting objects: read-only)** appears next to the bucket's name on the Buckets page.



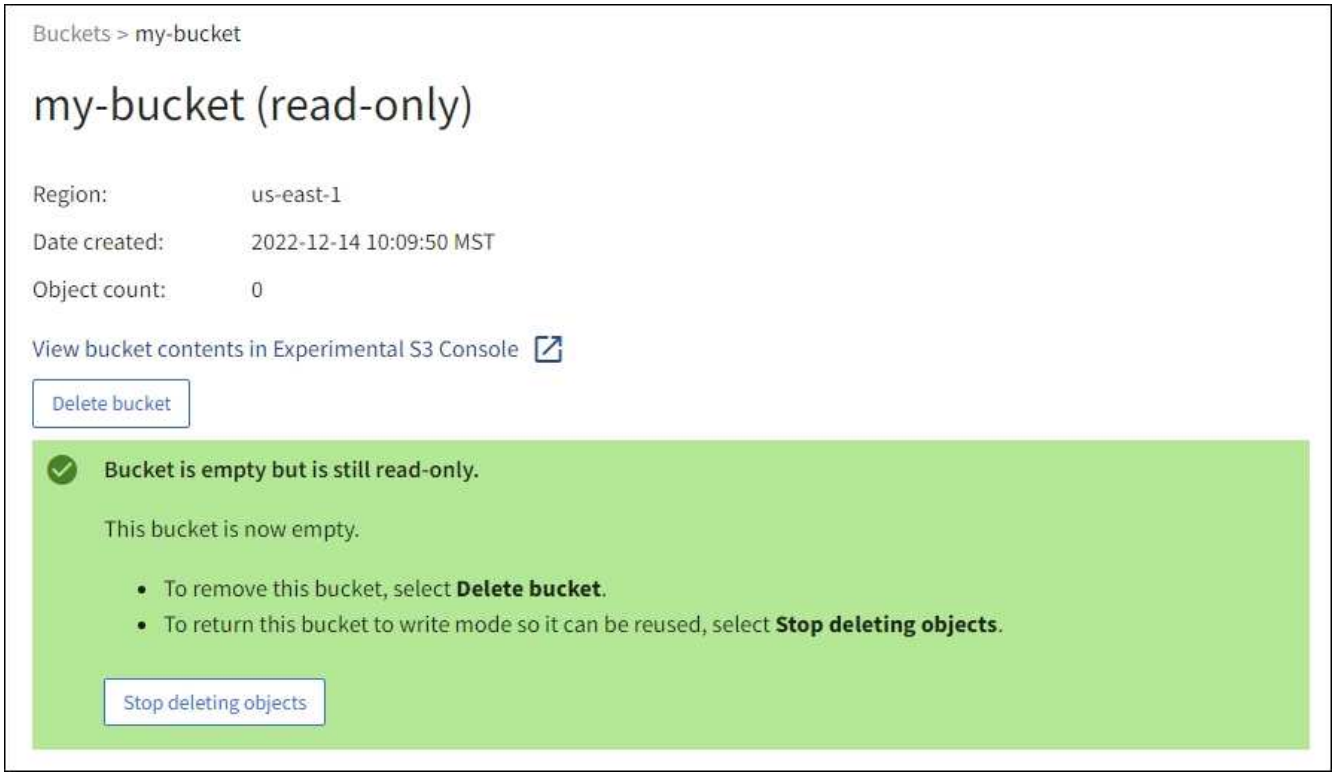
5. As required while the operation is running, select **Stop deleting objects** to halt the process. Then,

optionally, select **Delete objects in bucket** to resume the process.

When you select **Stop deleting objects**, the bucket is returned to write mode; however, you can't access or restore any objects that have been deleted.

6. Wait for the operation to complete.

   When the bucket is empty, the status banner is updated, but the bucket remains read only.



7. Do one of the following:
   - Exit the page to keep the bucket in read-only mode. For example, you might keep an empty bucket in read-only mode to reserve the bucket name for future use.
   - Delete the bucket. You can select **Delete bucket** to delete a single bucket or return the Buckets page and select **Actions** > **Delete** buckets to remove more than one bucket.

     If you are unable to delete a versioned bucket after all objects were deleted, delete markers might remain. To delete the bucket, you must remove all remaining delete markers.

   - Return the bucket to write mode and optionally reuse it for new objects. You can select **Stop deleting objects** for a single bucket or return to the Buckets page and select **Action** > **Stop deleting objects** for more than one bucket.

Delete S3 bucket

You can use the Tenant Manager to delete one or more S3 buckets that are empty.

Before you begin

- You are signed in to the Tenant Manager using a supported web browser.

- You belong to a user group that has the Manage all buckets or Root access permission. These permissions override the permissions settings in group or bucket policies.
- The buckets you want to delete are empty. If buckets you want to delete are *not* empty, delete objects from the bucket.

About this task

These instructions describe how to delete an S3 bucket using the Tenant Manager. You can also delete S3 buckets using the Tenant Management API or the S3 REST API.

You can't delete an S3 bucket if it contains objects, noncurrent object versions, or delete markers. For information about how S3 versioned objects are deleted, see How objects are deleted.

Steps

1. Select **View buckets** from the dashboard, or select **STORAGE (S3)** > **Buckets**.

   The Buckets page appears and shows all existing S3 buckets.

2. Use the **Actions** menu or the details page for a specific bucket.

   Actions menu

   a. Select the checkbox for each bucket you want to delete.

   b. Select **Actions** > **Delete buckets**.

   Details page

   a. Select a bucket name to display its details.

   b. Select **Delete bucket**.

3. When the confirmation dialog box appears, select **Yes**.

   StorageGRID confirms that each bucket is empty and then deletes each bucket. This operation might take a few minutes.

   If a bucket is not empty, an error message appears. You must delete all objects and any delete markers in the bucket before you can delete the bucket.

Use S3 Console

You can use S3 Console to view and manage the objects in an S3 bucket.

S3 Console allows you to:

- Upload, download, rename, copy, move, and delete objects
- View, revert, download, and delete object versions
- Search for objects by prefix
- Manage object tags
- View object metadata
- View, create, rename, copy, move, and delete folders

S3 Console provides an improved user experience for the most common cases. It is not designed to replace CLI or API operations in all situations.

If using S3 Console results in operations taking too long (for example, minutes or hours), consider:

- Reducing the number of selected objects
- Using non-graphical (API or CLI) methods to access your data

Before you begin

- You are signed in to the Tenant Manager using a supported web browser.
- If you want to manage objects, you belong to a user group that has the Root access permission. Alternatively, you belong to a user group that has the Use S3 Console tab permission and either the View all buckets permission or Manage all buckets permission. See Tenant management permissions.
- An S3 Group or Bucket policy has been configured for the user. See Use bucket and group access policies.
- You know the user's access key ID and secret access key. Optionally, you have a `.csv` file containing this information. See the instructions for creating access keys.

Steps

1. Select **STORAGE** > **Buckets** > ***bucket name***.
2. Select the S3 Console tab.
3. Paste the access key ID and secret access key into the fields. Otherwise, select **Upload access keys** and select your `.csv` file.
4. Select **Sign in**.
5. The table of bucket objects appears. You can manage objects as needed.

Additional information

- **Search by prefix**: The prefix search feature only searches for objects that begin with a specific word relative to the current folder. The search does not include objects that contain the word elsewhere. This rule also applies to objects within folders. For example, a search for `folder1/folder2/somefile-` would return objects that are within the `folder1/folder2/` folder and begin with the word `somefile-`.
- **Drag and drop**: You can drag and drop files from your computer's file manager to S3 Console. However, you cannot upload folders.
- **Operations on folders**: When you move, copy, or rename a folder, all objects in the folder are updated one at a time, which might take time.
- **Permanent deletion when bucket versioning is disabled**: When you overwrite or delete an object in a bucket with versioning disabled, the operation is permanent. See Change object versioning for a bucket.

Manage S3 platform services

S3 platform services

Platform services overview and considerations

Before implementing platform services, review the overview and considerations for using these services.

For information about S3, see Use S3 REST API.

Overview of platform services

StorageGRID platform services can help you implement a hybrid cloud strategy by allowing you to send event notifications and copies of S3 objects and object metadata to external destinations.

Because the target location for platform services is typically external to your StorageGRID deployment, platform services give you the power and flexibility that comes from using external storage resources, notification services, and search or analysis services for your data.

Any combination of platform services can be configured for a single S3 bucket. For example, you could configure both the CloudMirror service and notifications on a StorageGRID S3 bucket so that you can mirror specific objects to the Amazon Simple Storage Service (S3), while sending a notification about each such object to a third party monitoring application to help you track your AWS expenses.

> The use of platform services must be enabled for each tenant account by a StorageGRID administrator using the Grid Manager or the Grid Management API.

How platform services are configured

Platform services communicate with external endpoints that you configure using the Tenant Manager or the Tenant Management API. Each endpoint represents an external destination, such as a StorageGRID S3 bucket, an Amazon Web Services bucket, an Amazon SNS topic, or an Elasticsearch cluster hosted locally, on AWS, or elsewhere.

After you create an external endpoint, you can enable a platform service for a bucket by adding XML configuration to the bucket. The XML configuration identifies the objects that the bucket should act on, the action that the bucket should take, and the endpoint that the bucket should use for the service.

You must add separate XML configurations for each platform service that you want to configure. For example:

- If you want all objects whose keys start with `/images` to be replicated to an Amazon S3 bucket, you must add a replication configuration to the source bucket.
- If you also want to send notifications when these objects are stored to the bucket, you must add a notifications configuration.
- If you want to index the metadata for these objects, you must add the metadata notification configuration that is used to implement search integration.

The format for the configuration XML is governed by the S3 REST APIs used to implement StorageGRID platform services:

Platform service	S3 REST API	Refer to
CloudMirror replication	• GetBucketReplication • PutBucketReplication	• CloudMirror replication • Operations on buckets

Platform service	S3 REST API	Refer to
Notifications	• GetBucketNotificationConfiguration • PutBucketNotificationConfiguration	• Notifications • Operations on buckets
Search integration	• GET Bucket metadata notification configuration • PUT Bucket metadata notification configuration	• Search integration • StorageGRID custom operations

Considerations for using platform services

Consideration	Details
Destination endpoint monitoring	You must monitor the availability of each destination endpoint. If connectivity to the destination endpoint is lost for an extended period of time and a large backlog of requests exists, additional client requests (such as PUT requests) to StorageGRID will fail. You must retry these failed requests when the endpoint becomes reachable.
Destination endpoint throttling	StorageGRID software might throttle incoming S3 requests for a bucket if the rate at which the requests are being sent exceeds the rate at which the destination endpoint can receive the requests. Throttling only occurs when there is a backlog of requests waiting to be sent to the destination endpoint. The only visible effect is that the incoming S3 requests will take longer to execute. If you start to detect significantly slower performance, you should reduce the ingest rate or use an endpoint with higher capacity. If the backlog of requests continues to grow, client S3 operations (such as PUT requests) will eventually fail. CloudMirror requests are more likely to be affected by the performance of the destination endpoint because these requests typically involve more data transfer than search integration or event notification requests.
Ordering guarantees	StorageGRID guarantees ordering of operations on an object within a site. As long as all operations against an object are within the same site, the final object state (for replication) will always equal the state in StorageGRID. StorageGRID makes a best effort attempt to order requests when operations are made across StorageGRID sites. For example, if you write an object initially to site A and then later overwrite the same object at site B, the final object replicated by CloudMirror to the destination bucket is not guaranteed to be the newer object.

Consideration	Details
ILM-driven object deletions	To match the deletion behavior of the AWS CRR and Amazon Simple Notification Service, CloudMirror and event notification requests aren't sent when an object in the source bucket is deleted because of StorageGRID ILM rules. For example, no CloudMirror or event notifications requests are sent if an ILM rule deletes an object after 14 days. In contrast, search integration requests are sent when objects are deleted because of ILM.
Using Kafka endpoints	For Kafka endpoints, Mutual TLS is not supported. As a result, if you have `ssl.client.auth` set to `required` in your Kafka broker configuration, it might cause Kafka endpoint configuration issues. The authentication of Kafka endpoints uses the following authentication types. These types are different from those used for the authentication of other endpoints, such as Amazon SNS, and require username and password credentials. • SASL/PLAIN • SASL/SCRAM-SHA-256 • SASL/SCRAM-SHA-512 **Note:** Configured storage proxy settings do not apply to Kafka platform services endpoints.

Considerations for using CloudMirror replication service

Consideration	Details
Replication status	StorageGRID does not support the `x-amz-replication-status` header.
Object size	The maximum size for objects that can be replicated to a destination bucket by the CloudMirror replication service is 5 TiB, which is the same as the maximum *supported* object size. **Note**: The maximum *recommended* size for a single PutObject operation is 5 GiB (5,368,709,120 bytes). If you have objects that are larger than 5 GiB, use multipart upload instead.
Bucket versioning and version IDs	If the source S3 bucket in StorageGRID has versioning enabled, you should also enable versioning for the destination bucket. When using versioning, note that the ordering of object versions in the destination bucket is best effort and not guaranteed by the CloudMirror service, due to limitations in the S3 protocol. **Note**: Version IDs for the source bucket in StorageGRID aren't related to the version IDs for the destination bucket.

Consideration	Details
Tagging for object versions	The CloudMirror service does not replicate any PutObjectTagging or DeleteObjectTagging requests that supply a version ID, due to limitations in the S3 protocol. Because version IDs for the source and destination aren't related, there is no way to ensure that a tag update to a specific version ID will be replicated. In contrast, the CloudMirror service does replicate PutObjectTagging requests or DeleteObjectTagging requests that don't specify a version ID. These requests update the tags for the latest key (or the latest version if the bucket is versioned). Normal ingests with tags (not tagging updates) are also replicated.
Multipart uploads and `ETag` values	When mirroring objects that were uploaded using a multipart upload, the CloudMirror service does not preserve the parts. As a result, the `ETag` value for the mirrored object will be different than the `ETag` value of the original object.
Objects encrypted with SSE-C (server-side encryption with customer-provided keys)	The CloudMirror service does not support objects that are encrypted with SSE-C. If you attempt to ingest an object into the source bucket for CloudMirror replication and the request includes the SSE-C request headers, the operation fails.
Bucket with S3 Object Lock enabled	Replication is not supported for source or destination buckets with S3 Object Lock enabled.

Understand CloudMirror replication service

You can enable CloudMirror replication for an S3 bucket if you want StorageGRID to replicate specified objects added to the bucket to one or more external destination buckets.

For example, you might use CloudMirror replication to mirror specific customer records into Amazon S3 and then leverage AWS services to perform analytics on your data.

CloudMirror replication is not supported if the source bucket has S3 Object Lock enabled.

CloudMirror and ILM

CloudMirror replication operates independently of the grid's active ILM policies. The CloudMirror service replicates objects as they are stored to the source bucket and delivers them to the destination bucket as soon as possible. Delivery of replicated objects is triggered when object ingest succeeds.

CloudMirror and cross-grid replication

CloudMirror replication has important similarities and differences with the cross-grid replication feature. Refer to Compare cross-grid replication and CloudMirror replication.

CloudMirror and S3 buckets

CloudMirror replication is typically configured to use an external S3 bucket as a destination. However, you can also configure replication to use another StorageGRID deployment or any S3-compatible service.

Existing buckets

When you enable CloudMirror replication for an existing bucket, only the new objects added to that bucket are replicated. Any existing objects in the bucket aren't replicated. To force the replication of existing objects, you can update the existing object's metadata by performing an object copy.

> If you are using CloudMirror replication to copy objects to an Amazon S3 destination, be aware that Amazon S3 limits the size of user-defined metadata within each PUT request header to 2 KB. If an object has user-defined metadata greater than 2 KB, that object will not be replicated.

Multiple destination buckets

To replicate objects in a single bucket to multiple destination buckets, specify the destination for each rule in the replication configuration XML. You can't replicate an object to more than one bucket at the same time.

Versioned or unversioned buckets

You can configure CloudMirror replication on versioned or unversioned buckets. The destination buckets can be versioned or unversioned. You can use any combination of versioned and unversioned buckets. For example, you could specify a versioned bucket as the destination for an unversioned source bucket, or vice versa. You can also replicate between unversioned buckets.

Deletion, replication loops, and events

Deletion behavior

Is the same as the deletion behavior of the Amazon S3 service, Cross-Region Replication (CRR). Deleting an object in a source bucket never deletes a replicated object in the destination. If both source and destination buckets are versioned, the delete marker is replicated. If the destination bucket is not versioned, deleting an object in the source bucket doesn't replicate the delete marker to the destination bucket or delete the destination object.

Protection from replication loops

As objects are replicated to the destination bucket, StorageGRID marks them as "replicas." A destination StorageGRID bucket won't replicate objects marked as replicas again, protecting you from accidental replication loops. This replica marking is internal to StorageGRID and doesn't prevent you from leveraging AWS CRR when using an Amazon S3 bucket as the destination.

> The custom header used to mark a replica is `x-ntap-sg-replica`. This marking prevents a cascading mirror. StorageGRID does support a bidirectional CloudMirror between two grids.

Events in the destination bucket

The uniqueness and ordering of events in the destination bucket aren't guaranteed. More than one identical copy of a source object might be delivered to the destination as a result of operations taken to guarantee delivery success. In rare cases, when the same object is updated simultaneously from two or more different StorageGRID sites, the ordering of operations on the destination bucket might not match the ordering of events on the source bucket.

Understand notifications for buckets

You can enable event notification for an S3 bucket if you want StorageGRID to send notifications about specified events to a destination Kafka cluster or Amazon Simple Notification Service.

For example, you could configure alerts to be sent to administrators about each object added to a bucket, where the objects represent log files associated with a critical system event.

Event notifications are created at the source bucket as specified in the notification configuration and are delivered to the destination. If an event associated with an object succeeds, a notification about that event is created and queued for delivery.

The uniqueness and ordering of notifications aren't guaranteed. More than one notification of an event might be delivered to the destination as a result of operations taken to guarantee delivery success. And because delivery is asynchronous, the time ordering of notifications at the destination is not guaranteed to match the ordering of events on the source bucket, particularly for operations that originate from different StorageGRID sites. You can use the `sequencer` key in the event message to determine the order of events for a particular object, as described in Amazon S3 documentation.

StorageGRID event notifications follow the Amazon S3 API with some limitations.

- The following event types are supported:
  - s3:ObjectCreated:
  - s3:ObjectCreated:Put
  - s3:ObjectCreated:Post
  - s3:ObjectCreated:Copy
  - s3:ObjectCreated:CompleteMultipartUpload
  - s3:ObjectRemoved:
  - s3:ObjectRemoved:Delete
  - s3:ObjectRemoved:DeleteMarkerCreated
  - s3:ObjectRestore:Post
- Event notifications sent from StorageGRID use the standard JSON format but don't include some keys and use specific values for others, as shown in the table:

Key name	StorageGRID value
eventSource	`sgws:s3`
awsRegion	*not included*
x-amz-id-2	*not included*
arn	`urn:sgws:s3:::bucket_name`

Understand search integration service

You can enable search integration for an S3 bucket if you want to use an external search and data analysis service for your object metadata.

The search integration service is a custom StorageGRID service that automatically and asynchronously sends S3 object metadata to a destination endpoint whenever an object is created or deleted, or its metadata or tags are updated. You can then use sophisticated search, data analysis, visualization, or machine learning tools

provided by the destination service to search, analyze, and gain insights from your object data.

For example, you could configure your buckets to send S3 object metadata to a remote Elasticsearch service. You could then use Elasticsearch to perform searches across buckets, and perform sophisticated analyses of patterns present in your object metadata.

Although Elasticsearch integration can be configured on a bucket with S3 Object Lock enabled, the S3 Object Lock metadata (including Retain Until Date and Legal Hold status) of the objects will not be included in the metadata sent to Elasticsearch.

> Because the search integration service causes object metadata to be sent to a destination, its configuration XML is referred to as "*metadata* notification configuration XML." This configuration XML is different from the "notification configuration XML" used to enable *event* notifications.

Search integration and S3 buckets

You can enable the search integration service for any versioned or unversioned bucket. Search integration is configured by associating metadata notification configuration XML with the bucket that specifies which objects to act on and the destination for the object metadata.

Metadata notifications are generated in the form of a JSON document named with the bucket name, object name, and version ID, if any. Each metadata notification contains a standard set of system metadata for the object in addition to all of the object's tags and user metadata.

> For tags and user metadata, StorageGRID passes dates and numbers to Elasticsearch as strings or as S3 event notifications. To configure Elasticsearch to interpret these strings as dates or numbers, follow the Elasticsearch instructions for dynamic field mapping and for mapping date formats. You must enable the dynamic field mappings on the index before you configure the search integration service. After a document is indexed, you can't edit the document's field types in the index.

Search notifications

Metadata notifications are generated and queued for delivery whenever:

- An object is created.
- An object is deleted, including when objects are deleted as a result of the operation of the grid's ILM policy.
- Object metadata or tags are added, updated, or deleted. The complete set of metadata and tags is always sent on update — not just the changed values.

After you add metadata notification configuration XML to a bucket, notifications are sent for any new objects that you create and for any objects that you modify by updating its data, user metadata, or tags. However, notifications aren't sent for any objects that were already in the bucket. To ensure that object metadata for all objects in the bucket is sent to the destination, you should do either of the following:

- Configure the search integration service immediately after creating the bucket and before adding any objects.
- Perform an action on all objects already in the bucket that will trigger a metadata notification message to be sent to the destination.

Search integration service and Elasticsearch

The StorageGRID search integration service supports an Elasticsearch cluster as a destination. As with the other platform services, the destination is specified in the endpoint whose URN is used in the configuration XML for the service. Use the NetApp Interoperability Matrix Tool to determine the supported versions of Elasticsearch.

Manage platform services endpoints

Configure platform services endpoints

Before you can configure a platform service for a bucket, you must configure at least one endpoint to be the destination for the platform service.

Access to platform services is enabled on a per-tenant basis by a StorageGRID administrator. To create or use a platform services endpoint, you must be a tenant user with Manage endpoints or Root access permission, in a grid whose networking has been configured to allow Storage Nodes to access external endpoint resources. For a single tenant, you can configure a maximum of 500 platform services endpoints. Contact your StorageGRID administrator for more information.

What is a platform services endpoint?

A platform services endpoint specifies the information that StorageGRID needs to access the external destination.

For example, if you want to replicate objects from a StorageGRID bucket to an Amazon S3 bucket, you create a platform services endpoint that includes the information and credentials StorageGRID needs to access the destination bucket on Amazon.

Each type of platform service requires its own endpoint, so you must configure at least one endpoint for each platform service you plan to use. After defining a platform services endpoint, you use the endpoint's URN as the destination in the configuration XML used to enable the service.

You can use the same endpoint as the destination for more than one source bucket. For example, you could configure several source buckets to send object metadata to the same search integration endpoint so that you can perform searches across multiple buckets. You can also configure a source bucket to use more than one endpoint as a target, which enables you to do things like send notifications about object creation to one Amazon Simple Notification Service (Amazon SNS) topic and notifications about object deletion to a second Amazon SNS topic.

Endpoints for CloudMirror replication

StorageGRID supports replication endpoints that represent S3 buckets. These buckets might be hosted on Amazon Web Services, the same or a remote StorageGRID deployment, or another service.

Endpoints for notifications

StorageGRID supports Amazon SNS and Kafka endpoints. Simple Queue Service (SQS) or AWS Lambda endpoints aren't supported.

For Kafka endpoints, Mutual TLS is not supported. As a result, if you have `ssl.client.auth` set to `required` in your Kafka broker configuration, it might cause Kafka endpoint configuration issues.

Endpoints for the search integration service

StorageGRID supports search integration endpoints that represent Elasticsearch clusters. These Elasticsearch clusters can be in a local data center or hosted in an AWS cloud or elsewhere.

The search integration endpoint refers to a specific Elasticsearch index and type. You must create the index in Elasticsearch before creating the endpoint in StorageGRID, or endpoint creation will fail. You don't need to create the type before creating the endpoint. StorageGRID will create the type if required when it sends object metadata to the endpoint.

Related information

Administer StorageGRID

Specify URN for platform services endpoint

When you create a platform services endpoint, you must specify a Unique Resource Name (URN). You will use the URN to reference the endpoint when you create a configuration XML for the platform service. The URN for each endpoint must be unique.

StorageGRID validates platform services endpoints as you create them. Before you create a platform services endpoint, confirm that the resource specified in the endpoint exists and that it can be reached.

URN elements

The URN for a platform services endpoint must start with either `arn:aws` or `urn:mysite`, as follows:

- If the service is hosted on Amazon Web Services (AWS), use `arn:aws`
- If the service is hosted on Google Cloud Platform (GCP), use `arn:aws`
- If the service is hosted locally, use `urn:mysite`

For example, if you are specifying the URN for a CloudMirror endpoint hosted on StorageGRID, the URN might begin with `urn:sgws`.

The next element of the URN specifies the type of platform service, as follows:

Service	Type
CloudMirror replication	`s3`
Notifications	`sns` or `kafka`
Search integration	`es`

For example, to continue specifying the URN for a CloudMirror endpoint hosted on StorageGRID, you would add `s3` to get `urn:sgws:s3`.

The final element of the URN identifies the specific target resource at the destination URI.

Service	Specific resource
CloudMirror replication	`bucket-name`
Notifications	`sns-topic-name` or `kafka-topic-name`
Search integration	`domain-name/index-name/type-name` **Note:** If the Elasticsearch cluster is **not** configured to create indexes automatically, you must create the index manually before you create the endpoint.

URNs for services hosted on AWS and GCP

For AWS and GCP entities, the complete URN is a valid AWS ARN. For example:

- CloudMirror replication:

```
arn:aws:s3:::bucket-name
```

- Notifications:

```
arn:aws:sns:region:account-id:topic-name
```

- Search integration:

```
arn:aws:es:region:account-id:domain/domain-name/index-name/type-name
```

> For an AWS search integration endpoint, the `domain-name` must include the literal string `domain/`, as shown here.

URNs for locally-hosted services

When using locally-hosted services instead of cloud services, you can specify the URN in any way that creates a valid and unique URN, as long as the URN includes the required elements in the third and final positions. You can leave the elements indicated by optional blank, or you can specify them in any way that helps you identify the resource and make the URN unique. For example:

- CloudMirror replication:

```
urn:mysite:s3:optional:optional:bucket-name
```

For a CloudMirror endpoint hosted on StorageGRID, you can specify a valid URN that begins with `urn:sgws`:

```
urn:sgws:s3:optional:optional:bucket-name
```

- Notifications:

  Specify an Amazon Simple Notification Service endpoint:

  ```
  urn:mysite:sns:optional:optional:sns-topic-name
  ```

  Specify a Kafka endpoint:

  ```
  urn:mysite:kafka:optional:optional:kafka-topic-name
  ```

- Search integration:

  ```
  urn:mysite:es:optional:optional:domain-name/index-name/type-name
  ```

  > For locally-hosted search integration endpoints, the `domain-name` element can be any string as long as the URN of the endpoint is unique.

Create platform services endpoint

You must create at least one endpoint of the correct type before you can enable a platform service.

Before you begin

- You are signed in to the Tenant Manager using a supported web browser.
- Platform services were enabled for your tenant account by a StorageGRID administrator.
- You belong to a user group that has the Manage endpoints or Root access permission.
- The resource referenced by the platform services endpoint have been created:
  - CloudMirror replication: S3 bucket
  - Event notification: Amazon Simple Notification Service (Amazon SNS) or Kafka topic
  - Search notification: Elasticsearch index, if the destination cluster is not configured to automatically create indexes.
- You have the information about the destination resource:
  - Host and port for the Uniform Resource Identifier (URI)

    > If you plan to use a bucket hosted on a StorageGRID system as an endpoint for CloudMirror replication, contact the grid administrator to determine the values you need to enter.

  - Unique Resource Name (URN)

Specify URN for platform services endpoint

- Authentication credentials (if required):

  Search integration endpoints

  For search integration endpoints, you can use the following credentials:

  - Access Key: Access key ID and secret access key
  - Basic HTTP: Username and password

  CloudMirror replication endpoints

  For CloudMirror replication endpoints, you can use the following credentials:

  - Access Key: Access key ID and secret access key
  - CAP (C2S Access Portal): Temporary credentials URL, server and client certificates, client keys, and an optional client private key passphrase.

  Amazon SNS endpoints

  For Amazon SNS endpoints, you can use the following credentials:

  - Access Key: Access key ID and secret access key

  Kafka endpoints

  For Kafka endpoints, you can use the following credentials:

  - SASL/PLAIN: Username and password
  - SASL/SCRAM-SHA-256: Username and password
  - SASL/SCRAM-SHA-512: Username and password

- Security certificate (if using a custom CA certificate)

- If the Elasticsearch security features are enabled, you have the monitor cluster privilege for connectivity testing, and either the write index privilege or both the index and delete index privileges for document updates.

Steps

1. Select **STORAGE (S3)** > **Platform services endpoints**. The Platform services endpoints page appears.
2. Select **Create endpoint**.
3. Enter a display name to briefly describe the endpoint and its purpose.

   The type of platform service that the endpoint supports is shown beside the endpoint name when it is listed on the Endpoints page, so you don't need to include that information in the name.

4. In the **URI** field, specify the Unique Resource Identifier (URI) of the endpoint.

   Use one of the following formats:

```
https://host:port
http://host:port
```

If you don't specify a port, the following default ports are used:

- Port 443 for HTTPS URIs and port 80 for HTTP URIs (most endpoints)
- Port 9092 for HTTPS and HTTP URIs (Kafka endpoints only)

For example, the URI for a bucket hosted on StorageGRID might be:

```
https://s3.example.com:10443
```

In this example, `s3.example.com` represents the DNS entry for the virtual IP (VIP) of the StorageGRID high availability (HA) group, and `10443` represents the port defined in the load balancer endpoint.

> Whenever possible, you should connect to an HA group of load-balancing nodes to avoid a single point of failure.

Similarly, the URI for a bucket hosted on AWS might be:

```
https://s3-aws-region.amazonaws.com
```

> If the endpoint is used for the CloudMirror replication service, don't include the bucket name in the URI. You include the bucket name in the **URN** field.

5. Enter the Unique Resource Name (URN) for the endpoint.

> You can't change an endpoint's URN after the endpoint has been created.

6. Select **Continue**.
7. Select a value for **Authentication type**.

Search integration endpoints

Enter or upload the credentials for a search integration endpoint.

The credentials that you supply must have write permissions for the destination resource.

Authentication type	Description	Credentials
Anonymous	Provides anonymous access to the destination. Only works for endpoints that have security disabled.	No authentication.
Access Key	Uses AWS-style credentials to authenticate connections with the destination.	• Access key ID • Secret access key
Basic HTTP	Uses a username and password to authenticate connections to the destination.	• Username • Password

CloudMirror replication endpoints

Enter or upload the credentials for a CloudMirror replication endpoint.

The credentials that you supply must have write permissions for the destination resource.

Authentication type	Description	Credentials
Anonymous	Provides anonymous access to the destination. Only works for endpoints that have security disabled.	No authentication.
Access Key	Uses AWS-style credentials to authenticate connections with the destination.	• Access key ID • Secret access key
CAP (C2S Access Portal)	Uses certificates and keys to authenticate connections to the destination.	• Temporary credentials URL • Server CA certificate (PEM file upload) • Client certificate (PEM file upload) • Client private key (PEM file upload, OpenSSL encrypted format or unencrypted private key format) • Client private key passphrase (optional)

Amazon SNS endpoints

Enter or upload the credentials for an Amazon SNS endpoint.

The credentials that you supply must have write permissions for the destination resource.

Authentication type	Description	Credentials
Anonymous	Provides anonymous access to the destination. Only works for endpoints that have security disabled.	No authentication.
Access Key	Uses AWS-style credentials to authenticate connections with the destination.	• Access key ID • Secret access key

Kafka endpoints

Enter or upload the credentials for a Kafka endpoint.

The credentials that you supply must have write permissions for the destination resource.

Authentication type	Description	Credentials
Anonymous	Provides anonymous access to the destination. Only works for endpoints that have security disabled.	No authentication.
SASL/PLAIN	Uses a username and password with plain text to authenticate connections to the destination.	• Username • Password
SASL/SCRAM-SHA-256	Uses a username and password using a challenge-response protocol and SHA-256 hashing to authenticate connections to the destination.	• Username • Password
SASL/SCRAM-SHA-512	Uses a username and password using a challenge-response protocol and SHA-512 hashing to authenticate connections to the destination.	• Username • Password

Select **Use delegation taken authentication** if the username and password are derived from a delegation token that was obtained from a Kafka cluster.

8. Select **Continue**.
9. Select a radio button for **Verify server** to choose how TLS connection to the endpoint is verified.

Type of certificate verification	Description
Use custom CA certificate	Use a custom security certificate. If you select this setting, copy and paste the custom security certificate in the **CA Certificate** text box.

Type of certificate verification	Description
Use operating system CA certificate	Use the default Grid CA certificate installed on the operating system to secure connections.
Do not verify certificate	The certificate used for the TLS connection is not verified. This option is not secure.

10. Select **Test and create endpoint**.
    - A success message appears if the endpoint can be reached using the specified credentials. The connection to the endpoint is validated from one node at each site.
    - An error message appears if endpoint validation fails. If you need to modify the endpoint to correct the error, select **Return to endpoint details** and update the information. Then, select **Test and create endpoint**.

> Endpoint creation fails if platform services aren't enabled for your tenant account. Contact your StorageGRID administrator.

After you have configured an endpoint, you can use its URN to configure a platform service.

Related information

- Specify URN for platform services endpoint
- Configure CloudMirror replication
- Configure event notifications
- Configure search integration service

Test connection for platform services endpoint

If the connection to a platform service has changed, you can test the connection for the endpoint to validate that the destination resource exists and that it can be reached using the credentials you specified.

Before you begin

- You are signed in to the Tenant Manager using a supported web browser.
- You belong to a user group that has the Manage endpoints or Root access permission.

About this task

StorageGRID does not validate that the credentials have the correct permissions.

Steps

1. Select **STORAGE (S3)** > **Platform services endpoints**.

   The Platform services endpoints page appears and shows the list of platform services endpoints that have already been configured.

2. Select the endpoint whose connection you want to test.

   The endpoint details page appears.

3. Select **Test connection**.
   - A success message appears if the endpoint can be reached using the specified credentials. The connection to the endpoint is validated from one node at each site.
   - An error message appears if endpoint validation fails. If you need to modify the endpoint to correct the error, select **Configuration** and update the information. Then, select **Test and save changes**.

Edit platform services endpoint

You can edit the configuration for a platform services endpoint to change its name, URI, or other details. For example, you might need to update expired credentials or change the URI to point to a backup Elasticsearch index for failover. You can't change the URN for a platform services endpoint.

Before you begin

- You are signed in to the Tenant Manager using a supported web browser.
- You belong to a user group that has the Manage endpoints or Root access permission.

Steps

1. Select **STORAGE (S3)** > **Platform services endpoints**.

   The Platform services endpoints page appears and shows the list of platform services endpoints that have already been configured.

2. Select the endpoint you want to edit.

   The endpoint details page appears.

3. Select **Configuration**.
4. As needed, change the configuration of the endpoint.

   > You can't change an endpoint's URN after the endpoint has been created.

   a. To change the display name for the endpoint, select the edit icon .
   b. As needed, change the URI.
   c. As needed, change the authentication type.
      - For Access Key authentication, change the key as necessary by selecting **Edit S3 key** and pasting a new access key ID and secret access key. If you need to cancel your changes, select **Revert S3 key edit**.
      - For CAP (C2S Access Portal) authentication, change the temporary credentials URL or optional client private key passphrase and upload new certificate and key files as needed.

        > The Client private key must be in OpenSSL encrypted format or unencrypted private key format.

   d. As needed, change the method for verifying the server.
5. Select **Test and save changes**.
   - A success message appears if the endpoint can be reached using the specified credentials. The connection to the endpoint is verified from one node at each site.

- An error message appears if endpoint validation fails. Modify the endpoint to correct the error, and then select **Test and save changes**.

Delete platform services endpoint

You can delete an endpoint if you no longer want to use the associated platform service.

Before you begin

- You are signed in to the Tenant Manager using a supported web browser.
- You belong to a user group that has the Manage endpoints or Root access permission.

Steps

1. Select **STORAGE (S3)** > **Platform services endpoints**.

   The Platform services endpoints page appears and shows the list of platform services endpoints that have already been configured.

2. Select the checkbox for each endpoint you want to delete.

   If you delete a platform services endpoint that is in use, the associated platform service will be disabled for any buckets that use the endpoint. Any requests that have not yet been completed will be dropped. Any new requests will continue to be generated until you change your bucket configuration to no longer reference the deleted URN. StorageGRID will report these requests as unrecoverable errors.

3. Select **Actions** > **Delete endpoint**.

   A confirmation message appears.

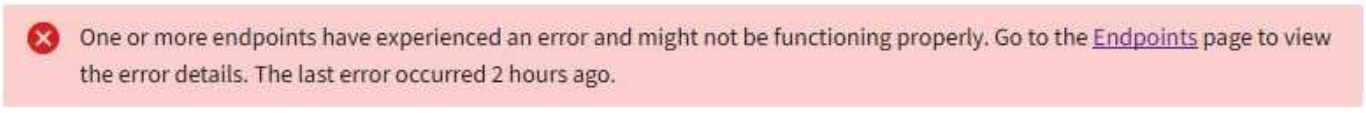
4. Select **Delete endpoint**.

Troubleshoot platform services endpoint errors

If an error occurs when StorageGRID attempts to communicate with a platform services endpoint, a message is displayed on the dashboard. On the Platform services endpoints page, the Last error column indicates how long ago the error occurred. No error is displayed if the permissions associated with an endpoint's credentials are incorrect.

Determine if error has occurred

If any platform services endpoint errors have occurred within the past 7 days, the Tenant Manager dashboard displays an alert message. You can go the Platform services endpoints page to see more details about the error.



The same error that appears on the dashboard also appears at the top of the Platform services endpoints page. To view a more detailed error message:

Steps

1. From the list of endpoints, select the endpoint that has the error.
2. On the endpoint details page, select **Connection**. This tab displays only the most recent error for an endpoint and indicates how long ago the error occurred. Errors that include the red X icon occurred within the past 7 days.

Check if error is still current

Some errors might continue to be shown in the **Last error** column even after they are resolved. To see if an error is current or to force the removal of a resolved error from the table:

Steps

1. Select the endpoint.

   The endpoint details page appears.

2. Select **Connection** > **Test connection**.

Selecting **Test connection** causes StorageGRID to validate that the platform services endpoint exists and that it can be reached with the current credentials. The connection to the endpoint is validated from one node at each site.

Resolve endpoint errors

You can use the **Last error** message on the endpoint details page to help determine what is causing the error. Some errors might require you to edit the endpoint to resolve the issue. For example, a CloudMirroring error can occur if StorageGRID is unable to access the destination S3 bucket because it does not have the correct access permissions or the access key has expired. The message is "Either the endpoint credentials or the destination access needs to be updated," and the details are "AccessDenied" or "InvalidAccessKeyId."

If you need to edit the endpoint to resolve an error, selecting **Test and save changes** causes StorageGRID to validate the updated endpoint and confirm that it can be reached with the current credentials. The connection to the endpoint is validated from one node at each site.

Steps

1. Select the endpoint.
2. On the endpoint details page, select **Configuration**.
3. Edit the endpoint configuration as needed.
4. Select **Connection** > **Test connection**.

Endpoint credentials with insufficient permissions

When StorageGRID validates a platform services endpoint, it confirms that the endpoint's credentials can be used to contact the destination resource and it does a basic permissions check. However, StorageGRID does not validate all of the permissions required for certain platform services operations. For this reason, if you receive an error when attempting to use a platform service (such as "403 Forbidden"), check the permissions associated with the endpoint's credentials.

Related information

- Administer StorageGRID > Troubleshoot platform services
- Create platform services endpoint
- Test connection for platform services endpoint

- Edit platform services endpoint

Configure CloudMirror replication

To enable CloudMirror replication for a bucket, you create and apply a valid bucket replication configuration XML.

Before you begin

- Platform services were enabled for your tenant account by a StorageGRID administrator.
- You have already created a bucket to act as the replication source.
- The endpoint that you intend to use as a destination for CloudMirror replication already exists, and you have its URN.
- You belong to a user group that has the Manage all buckets or Root access permission. These permissions override the permission settings in group or bucket policies when configuring the bucket using the Tenant Manager.

About this task

CloudMirror replication copies objects from a source bucket to a destination bucket that is specified in an endpoint.

For general information about bucket replication and how to configure it, see Amazon Simple Storage Service (S3) documentation: Replicating objects. For information about how StorageGRID implements GetBucketReplication, DeleteBucketReplication, and PutBucketReplication, see the Operations on buckets.

> CloudMirror replication has important similarities and differences with the cross-grid replication feature. To learn more, see Compare cross-grid replication and CloudMirror replication.

Note the following requirements and characteristics when configuring CloudMirror replication:

- When you create and apply a valid bucket replication configuration XML, it must use the URN of an S3 bucket endpoint for each destination.
- Replication is not supported for source or destination buckets with S3 Object Lock enabled.
- If you enable CloudMirror replication on a bucket that contains objects, new objects added to the bucket are replicated, but the existing objects in the bucket aren't replicated. You must update existing objects to trigger replication.
- If you specify a storage class in the replication configuration XML, StorageGRID uses that class when performing operations against the destination S3 endpoint. The destination endpoint must also support the specified storage class. Be sure to follow any recommendations provided by the destination system vendor.

Steps

1. Enable replication for your source bucket:
    - Use a text editor to create the replication configuration XML required to enable replication, as specified in the S3 replication API.
    - When configuring the XML:
        - Note that StorageGRID only supports V1 of the replication configuration. This means that StorageGRID does not support the use of the `Filter` element for rules, and follows V1 conventions for deletion of object versions. See the Amazon documentation on replication configuration for details.

- Use the URN of an S3 bucket endpoint as the destination.
- Optionally add the `<StorageClass>` element, and specify one of the following:
  - `STANDARD`: The default storage class. If you don't specify a storage class when you upload an object, the `STANDARD` storage class is used.
  - `STANDARD_IA`: (Standard - infrequent access.) Use this storage class for data that is accessed less frequently, but that still requires rapid access when needed.
  - `REDUCED_REDUNDANCY`: Use this storage class for noncritical, reproducible data that can be stored with less redundancy than the `STANDARD` storage class.
- If you specify a `Role` in the configuration XML it will be ignored. This value is not used by StorageGRID.

```
<ReplicationConfiguration>
    <Role></Role>
    <Rule>
        <Status>Enabled</Status>
        <Prefix>2020</Prefix>
        <Destination>
           <Bucket>urn:sgws:s3:::2017-records</Bucket>
           <StorageClass>STANDARD</StorageClass>
         </Destination>
     </Rule>
</ReplicationConfiguration>
```

2. Select **View buckets** from the dashboard, or select **STORAGE (S3)** > **Buckets**.
3. Select the name of the source bucket.

   The bucket details page appears.

4. Select **Platform services** > **Replication**.
5. Select the **Enable replication** checkbox.
6. Paste the replication configuration XML into the text box, and select **Save changes**.

   Platform services must be enabled for each tenant account by a StorageGRID administrator using the Grid Manager or Grid Management API. Contact your StorageGRID administrator if an error occurs when you save the configuration XML.

7. Verify that replication is configured correctly:
   a. Add an object to the source bucket that meets the requirements for replication as specified in the replication configuration.

      In the example shown earlier, objects that match the prefix "2020" are replicated.

   b. Confirm that the object has been replicated to the destination bucket.

      For small objects, replication happens quickly.

Related information

Create platform services endpoint

Configure event notifications

You enable notifications for a bucket by creating notification configuration XML and using the Tenant Manager to apply the XML to a bucket.

Before you begin

- Platform services were enabled for your tenant account by a StorageGRID administrator.
- You have already created a bucket to act as the source of notifications.
- The endpoint that you intend to use as a destination for event notifications already exists, and you have its URN.
- You belong to a user group that has the Manage all buckets or Root access permission. These permissions override the permission settings in group or bucket policies when configuring the bucket using the Tenant Manager.

About this task

You configure event notifications by associating notification configuration XML with a source bucket. The notification configuration XML follows S3 conventions for configuring bucket notifications, with the destination Kafka or Amazon SNS topic specified as the URN of an endpoint.

For general information about event notifications and how to configure them, refer to the Amazon documentation. For information about how StorageGRID implements the S3 bucket notification configuration API, refer to the instructions for implementing S3 client applications.

Note the following requirements and characteristics when configuring event notifications for a bucket:

- When you create and apply valid notification configuration XML, it must use the URN of an event notifications endpoint for each destination.
- Although event notification can be configured on a bucket with S3 Object Lock enabled, the S3 Object Lock metadata (including Retain Until Date and Legal Hold status) of the objects won't be included in the notification messages.
- After you configure event notifications, whenever a specified event occurs for an object in the source bucket, a notification is generated and sent to the Amazon SNS or Kafka topic used as the destination endpoint.
- If you enable event notifications for a bucket that contains objects, notifications are sent only for actions that are performed after the notification configuration is saved.

Steps

1. Enable notifications for your source bucket:
    - Use a text editor to create the notification configuration XML required to enable event notifications, as specified in the S3 notification API.
    - When configuring the XML, use the URN of an event notifications endpoint as the destination topic.

```
<NotificationConfiguration>
  <TopicConfiguration>
     <Id>Image-created</Id>
     <Filter>
        <S3Key>
          <FilterRule>
            <Name>prefix</Name>
            <Value>images/</Value>
          </FilterRule>
        </S3Key>
    </Filter>
    <Topic>arn:aws:sns:us-east-1:050340950352:sgws-topic</Topic>
    <Event>s3:ObjectCreated:*</Event>
   </TopicConfiguration>
 </NotificationConfiguration>
```

2. In the Tenant Manager, select **STORAGE (S3)** > **Buckets**.
3. Select the name of the source bucket.

   The bucket details page appears.

4. Select **Platform services** > **Event notifications**.
5. Select the **Enable event notifications** checkbox.
6. Paste the notification configuration XML into the text box, and select **Save changes**.

   > Platform services must be enabled for each tenant account by a StorageGRID administrator using the Grid Manager or Grid Management API. Contact your StorageGRID administrator if an error occurs when you save the configuration XML.

7. Verify that event notifications are configured correctly:
   a. Perform an action on an object in the source bucket that meets the requirements for triggering a notification as configured in the configuration XML.

      In the example, an event notification is sent whenever an object is created with the `images/` prefix.

   b. Confirm that a notification has been delivered to the destination Amazon SNS or Kafka topic.

      For example, if your destination topic is hosted on the Amazon SNS, you could configure the service to send you an email when the notification is delivered.

```
{
   "Records":[
      {
         "eventVersion":"2.0",
         "eventSource":"sgws:s3",
         "eventTime":"2017-08-08T23:52:38Z",
         "eventName":"ObjectCreated:Put",
         "userIdentity":{
            "principalId":"11111111111111111"
         },
         "requestParameters":{
            "sourceIPAddress":"193.51.100.20"
         },
         "responseElements":{
            "x-amz-request-id":"122047343"
         },
         "s3":{
            "s3SchemaVersion":"1.0",
            "configurationId":"Image-created",
            "bucket":{
               "name":"test1",
               "ownerIdentity":{
                  "principalId":"11111111111111111"
               },
               "arn":"arn:sgws:s3:::test1"
            },
            "object":{
               "key":"images/cat.jpg",
               "size":0,
               "eTag":"d41d8cd98f00b204e9800998ecf8427e",
               "sequencer":"14D90402421461C7"
            }
         }
      }
   ]
}
```


If the notification is received at the destination topic, you have successfully configured your source bucket for StorageGRID notifications.

Related information

Understand notifications for buckets

Use S3 REST API

Create platform services endpoint

Configure the search integration service

You enable search integration for a bucket by creating search integration XML and using the Tenant Manager to apply the XML to the bucket.

Before you begin

- Platform services were enabled for your tenant account by a StorageGRID administrator.
- You have already created an S3 bucket whose contents you want to index.
- The endpoint that you intend to use as a destination for the search integration service already exists, and you have its URN.
- You belong to a user group that has the Manage all buckets or Root access permission. These permissions override the permission settings in group or bucket policies when configuring the bucket using the Tenant Manager.

About this task

After you configure the search integration service for a source bucket, creating an object or updating an object's metadata or tags triggers object metadata to be sent to the destination endpoint.

If you enable the search integration service for a bucket that already contains objects, metadata notifications aren't automatically sent for existing objects. Update these existing objects to ensure that their metadata is added to the destination search index.

Steps

1. Enable search integration for a bucket:
   - Use a text editor to create the metadata notification XML required to enable search integration.
   - When configuring the XML, use the URN of a search integration endpoint as the destination.

     Objects can be filtered on the prefix of the object name. For example, you could send metadata for objects with the prefix `images` to one destination, and metadata for objects with the prefix `videos` to another. Configurations that have overlapping prefixes aren't valid, and are rejected when they're submitted. For example, a configuration that includes one rule for objects with the prefix `test` and a second rule for objects with the prefix `test2` is not allowed.

     As needed, refer to the examples for the metadata configuration XML.

     ```
     <MetadataNotificationConfiguration>
      <Rule>
         <Status>Enabled</Status>
         <Prefix></Prefix>
         <Destination>
            <Urn>/Urn>
            </Destination>
      </Rule>
     </MetadataNotificationConfiguration>
     ```

     Elements in the metadata notification configuration XML:

Name	Description	Required
MetadataNotificationConfiguration	Container tag for rules used to specify the objects and destination for metadata notifications. Contains one or more Rule elements.	Yes
Rule	Container tag for a rule that identifies the objects whose metadata should be added to a specified index. Rules with overlapping prefixes are rejected. Included in the MetadataNotificationConfiguration element.	Yes
ID	Unique identifier for the rule. Included in the Rule element.	No
Status	Status can be 'Enabled' or 'Disabled'. No action is taken for rules that are disabled. Included in the Rule element.	Yes
Prefix	Objects that match the prefix are affected by the rule, and their metadata is sent to the specified destination. To match all objects, specify an empty prefix. Included in the Rule element.	Yes
Destination	Container tag for the destination of a rule. Included in the Rule element.	Yes

Name	Description	Required
Urn	URN of the destination where object metadata is sent. Must be the URN of a StorageGRID endpoint with the following properties: • `es` must be the third element. • The URN must end with the index and type where the metadata is stored, in the form `domain-name/myindex/mytype`. Endpoints are configured using the Tenant Manager or Tenant Management API. They take the following form: • `arn:aws:es:region:account-ID:domain/mydomain/myindex/mytype` • `urn:mysite:es:::mydomain/myindex/mytype` The endpoint must be configured before the configuration XML is submitted, or configuration will fail with a 404 error. URN is included in the Destination element.	Yes

2. In the Tenant Manager select **STORAGE (S3)** > **Buckets**.
3. Select the name of the source bucket.

   The bucket details page appears.

4. Select **Platform services** > **Search integration**
5. Select the **Enable search integration** checkbox.
6. Paste the metadata notification configuration into the text box, and select **Save changes**.

   Platform services must be enabled for each tenant account by a StorageGRID administrator using the Grid Manager or Management API. Contact your StorageGRID administrator if an error occurs when you save the configuration XML.

7. Verify that the search integration service is configured correctly:
   a. Add an object to the source bucket that meets the requirements for triggering a metadata notification as specified in the configuration XML.

      In the example shown earlier, all objects added to the bucket trigger a metadata notification.

   b. Confirm that a JSON document that contains the object's metadata and tags was added to the search index specified in the endpoint.

After you finish

As necessary, you can disable search integration for a bucket using either of the following methods:

- Select **STORAGE (S3)** > **Buckets** and clear the **Enable search integration** checkbox.

- If you are using the S3 API directly, use a DELETE Bucket metadata notification request. See the instructions for implementing S3 client applications.

Example: Metadata notification configuration that applies to all objects

In this example, object metadata for all objects is sent to the same destination.

```
<MetadataNotificationConfiguration>
    <Rule>
        <ID>Rule-1</ID>
        <Status>Enabled</Status>
        <Prefix></Prefix>
        <Destination>
           <Urn>urn:myes:es:::sgws-notifications/test1/all</Urn>
        </Destination>
    </Rule>
</MetadataNotificationConfiguration>
```

Example: Metadata notification configuration with two rules

In this example, object metadata for objects that match the prefix `/images` is sent to one destination, while object metadata for objects that match the prefix `/videos` is sent to a second destination.

```
<MetadataNotificationConfiguration>
    <Rule>
        <ID>Images-rule</ID>
        <Status>Enabled</Status>
        <Prefix>/images</Prefix>
        <Destination>
           <Urn>arn:aws:es:us-east-1:3333333:domain/es-
domain/graphics/imagetype</Urn>
        </Destination>
    </Rule>
    <Rule>
        <ID>Videos-rule</ID>
        <Status>Enabled</Status>
        <Prefix>/videos</Prefix>
        <Destination>
           <Urn>arn:aws:es:us-west-1:22222222:domain/es-
domain/graphics/videotype</Urn>
        </Destination>
    </Rule>
</MetadataNotificationConfiguration>
```

Metadata notification format

When you enable the search integration service for a bucket, a JSON document is generated and sent to the destination endpoint each time object metadata or tags are added, updated, or deleted.

This example shows an example of the JSON that could be generated when an object with the key `SGWS/Tagging.txt` is created in a bucket named `test`. The `test` bucket is not versioned, so the `versionId` tag is empty.

```
{
  "bucket": "test",
  "key": "SGWS/Tagging.txt",
  "versionId": "",
  "accountId": "86928401983529626822",
  "size": 38,
  "md5": "3d6c7634a85436eee06d43415012855",
  "region":"us-east-1",
  "metadata": {
    "age": "25"
  },
  "tags": {
    "color": "yellow"
  }
}
```

Fields included in the JSON document

The document name includes the bucket name, object name, and version ID if present.

Bucket and object information

`bucket`: Name of the bucket

`key`: Object key name

`versionID`: Object version, for objects in versioned buckets

`region`: Bucket region, for example `us-east-1`

System metadata

`size`: Object size (in bytes) as visible to an HTTP client

`md5`: Object hash

User metadata

`metadata`: All user metadata for the object, as key-value pairs

`key:value`

Tags

`tags`: All object tags defined for the object, as key-value pairs

```
key:value
```

How to view results in Elasticsearch

For tags and user metadata, StorageGRID passes dates and numbers to Elasticsearch as strings or as S3 event notifications. To configure Elasticsearch to interpret these strings as dates or numbers, follow the Elasticsearch instructions for dynamic field mapping and for mapping date formats. Enable the dynamic field mappings on the index before you configure the search integration service. After a document is indexed, you can't edit the document's field types in the index.

Copyright information

Trademark information