Manage S3 tenant accounts

StorageGRID

NetApp
November 07, 2024

Table of Contents

Manage S3 tenant accounts

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.

About this task

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 in the S3 tenant account.

What you'll need

- You must be signed in to the Tenant Manager using a supported web browser.
- You must have the Manage Your Own S3 Credentials permission. See Tenant management permissions.

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, do not create more keys than you need, and delete the keys you are not 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 do not need to periodically create new keys, you do not have to set an expiration time for your keys. If you decide later to create new keys, delete the old keys manually.

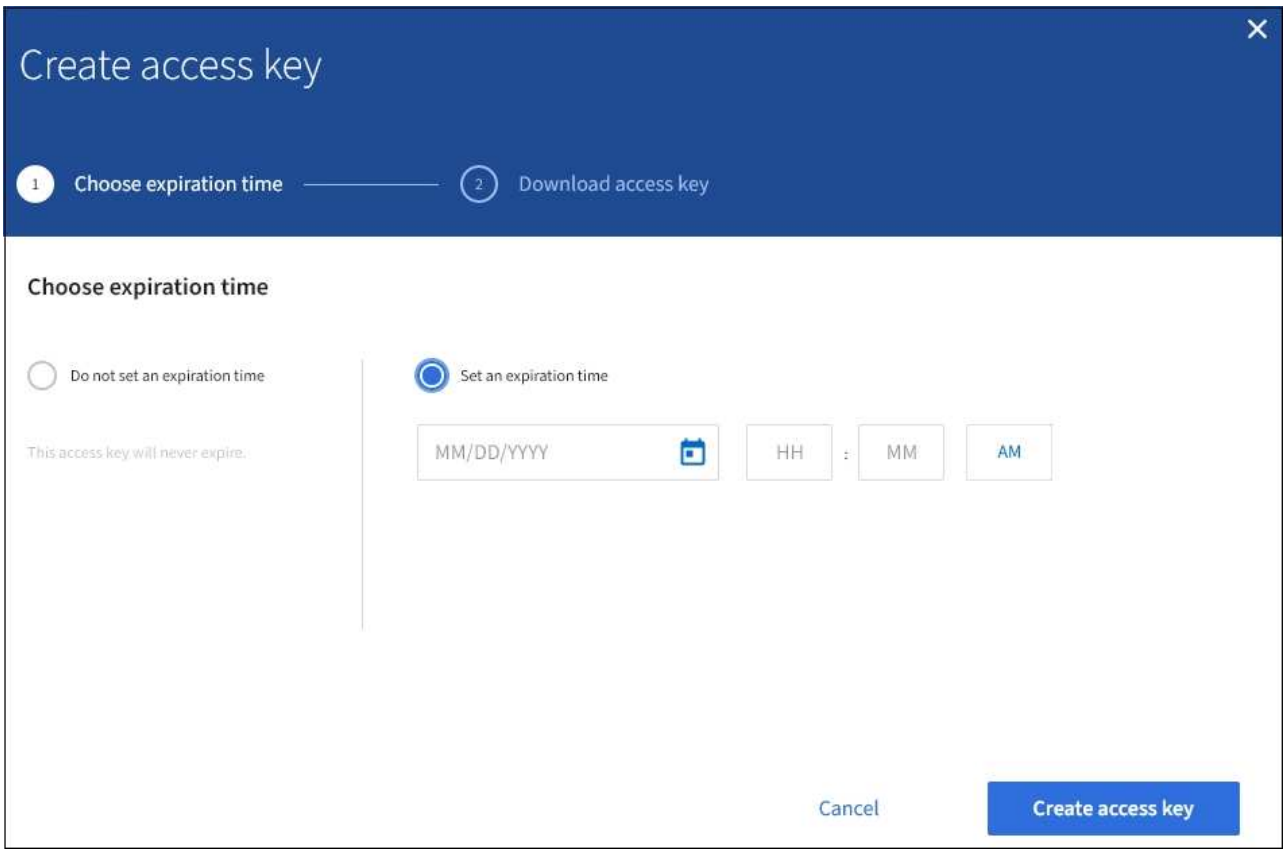
The S3 buckets and objects belonging to your account can be accessed using the access key ID and secret access key displayed for your account in the Tenant Manager. For this reason, protect access keys as you would a password. Rotate access keys on a regular basis, remove any unused keys from your account, 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.

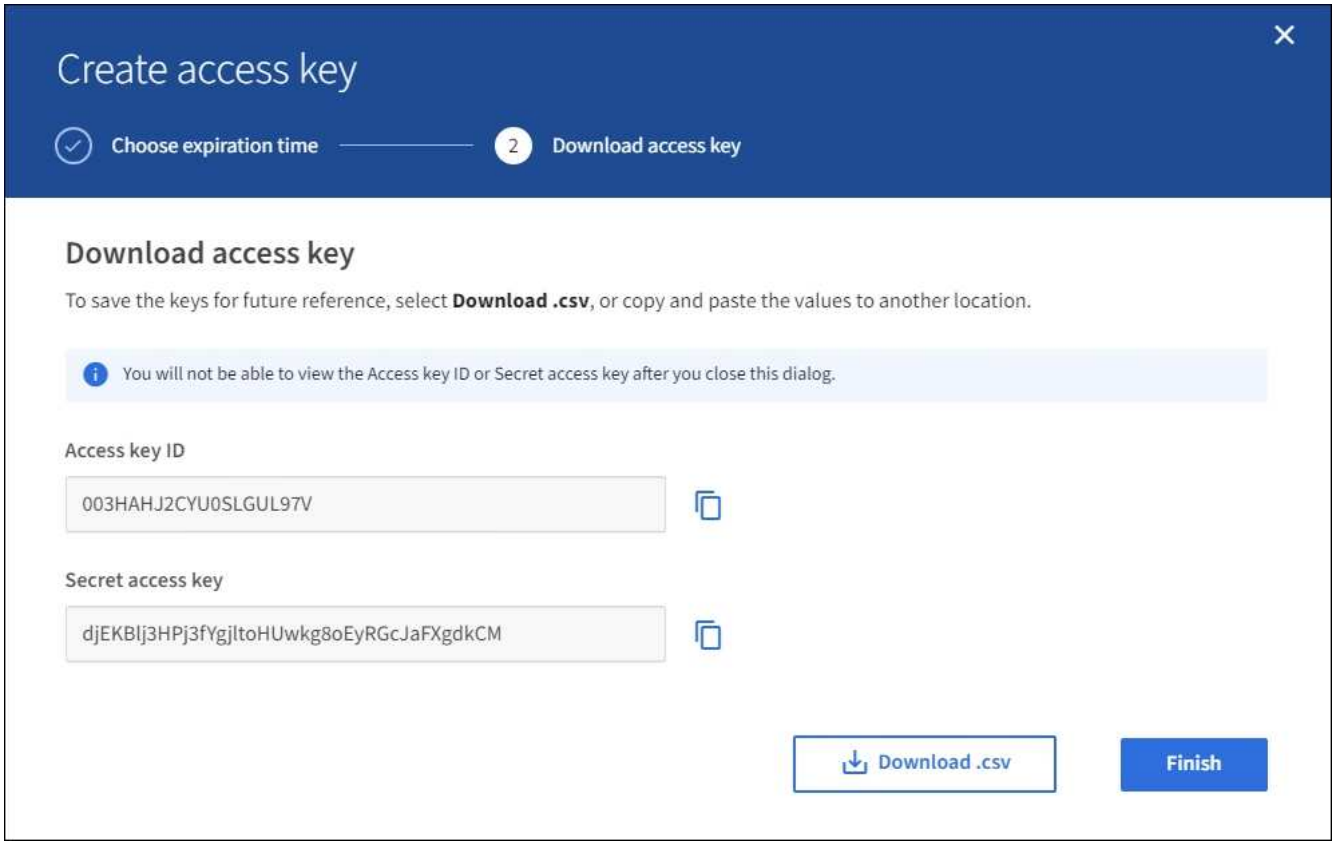


4. Select **Create access key**.

   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.

   Do not close this dialog box until you have copied or downloaded this information. You cannot 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. Changes might take up to 15 minutes to take effect because of caching.

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.

What you'll need

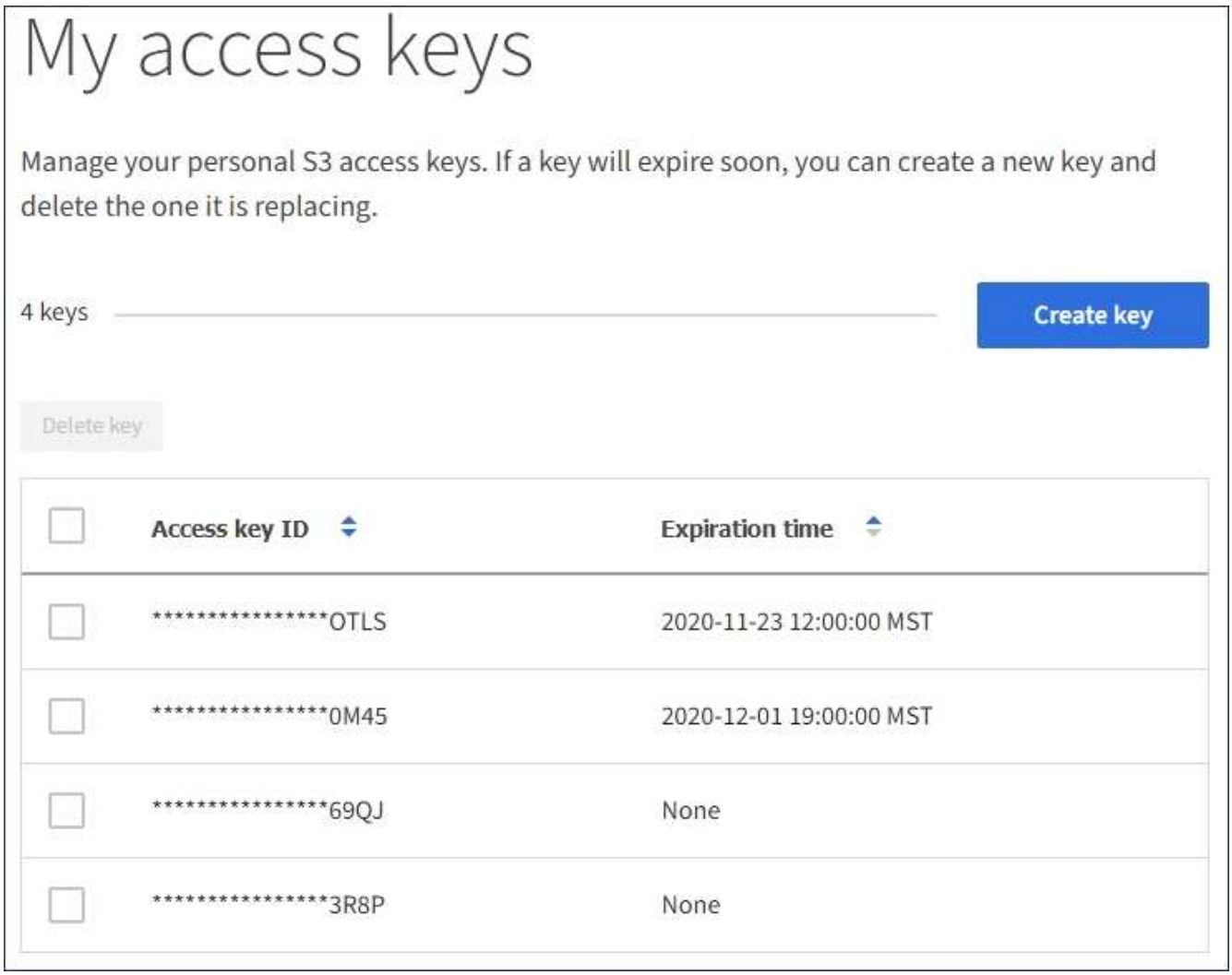
- You must be signed in to the Tenant Manager using a supported web browser.
- You must have the Manage Your Own S3 Credentials permission.

> 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**.

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



2. Sort the keys by **Expiration time** or **Access key ID**.
3. As needed, create new keys and manually delete 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.

Related information

Create your own S3 access keys

Delete your own S3 access keys

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.

What you'll need

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

- You must have the Manage Your Own S3 Credentials permission. See Tenant management permissions.

> 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.

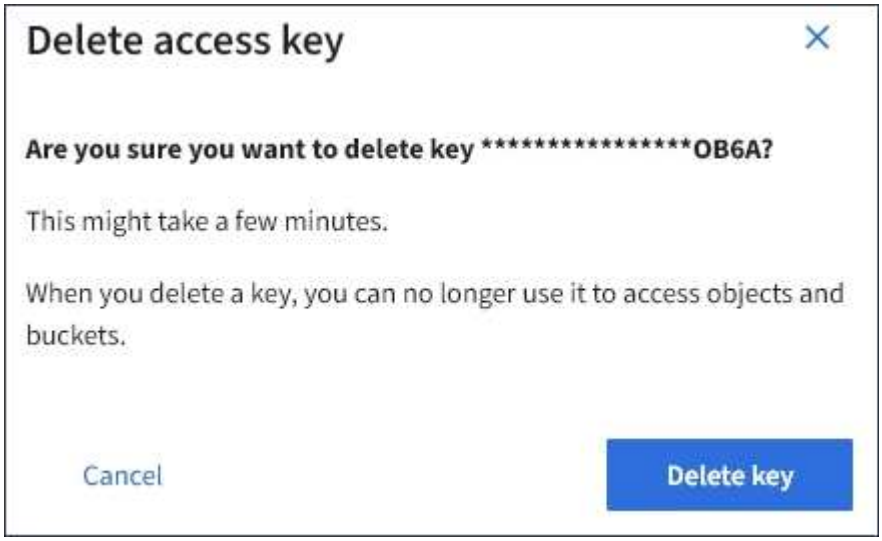
Steps

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

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

2. Select the check box for each access key you want to remove.
3. Select **Delete key**.

   A confirmation dialog box appears.



4. Select **Delete key**.

   A confirmation message appears in the upper right corner of the page. Changes might take up to 15 minutes to take effect because of caching.

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.

What you'll need

- You must be signed in to the Tenant Manager using a supported web browser.
- You must have 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, do not create more keys than the user needs, and delete the keys that are not 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 do not need to periodically create new keys, you do not have to set an expiration time for the keys. If you decide later to create new keys, delete the old keys manually.

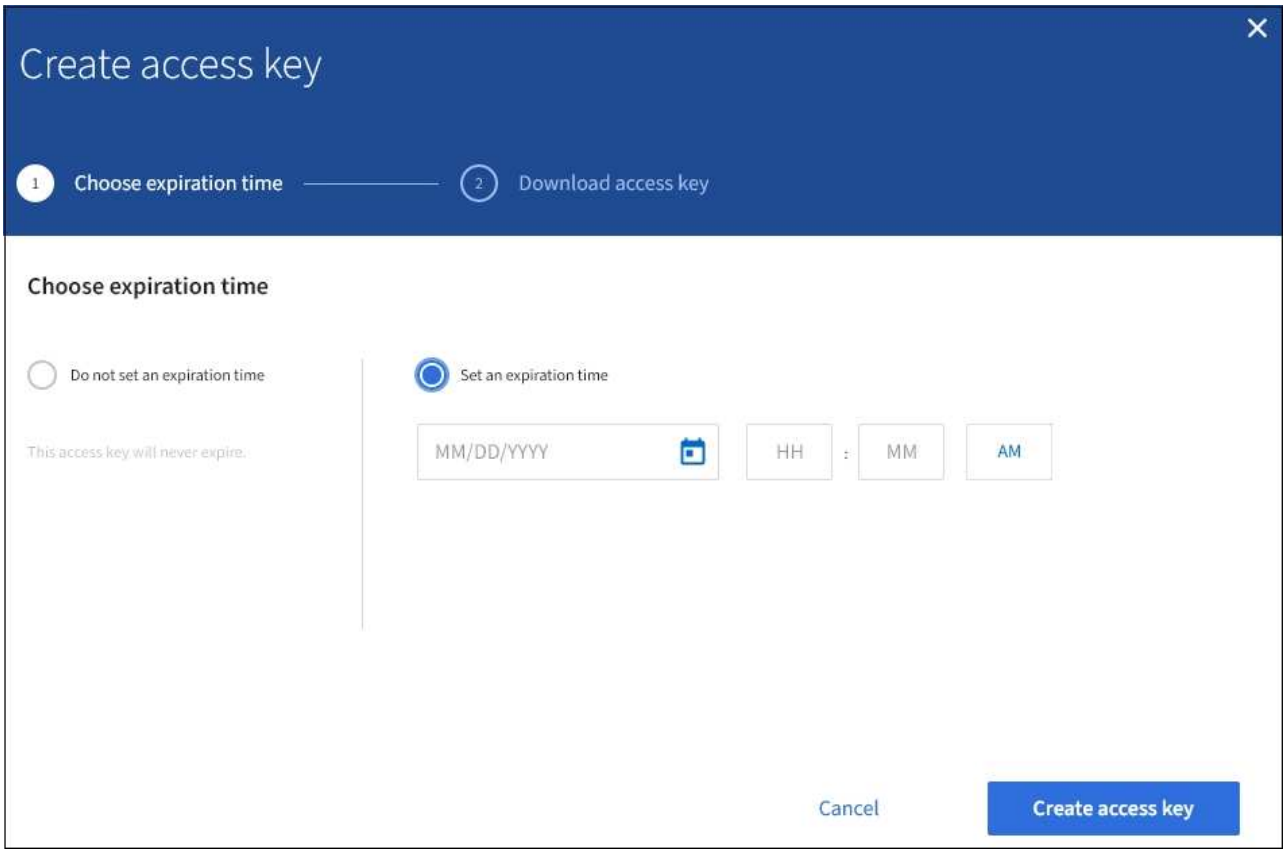
> The S3 buckets and objects belonging to a user can be accessed using the access key ID and secret access key displayed for that user in the Tenant Manager. For this reason, protect access keys as you would a password. Rotate access keys on a regular basis, remove any unused keys from the account, and 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 **Do not 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.

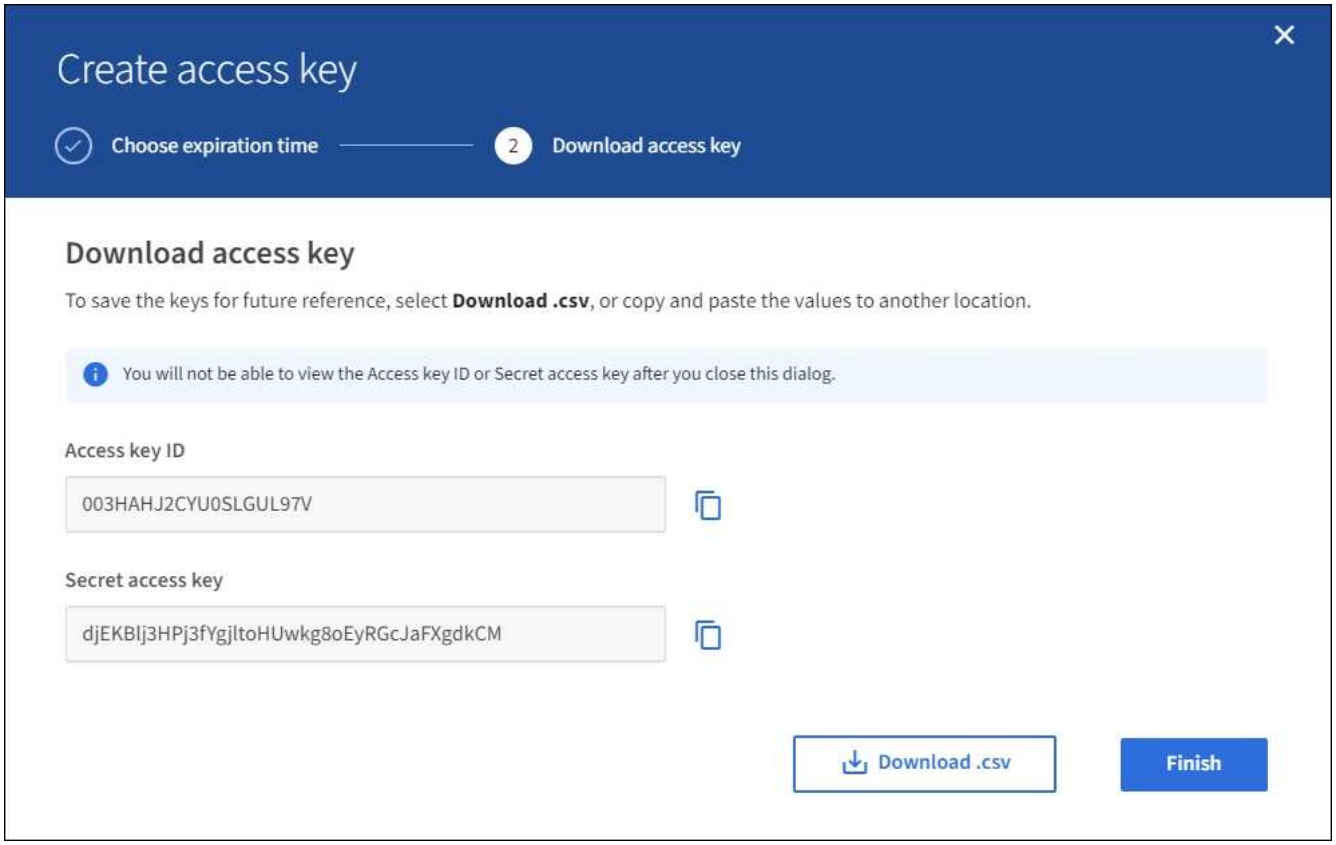


5. Select **Create access key**.

   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.

   Do not close this dialog box until you have copied or downloaded this information. You cannot 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. Changes might take up to 15 minutes to take effect because of caching.

Related information

Tenant management permissions

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.

What you'll need

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

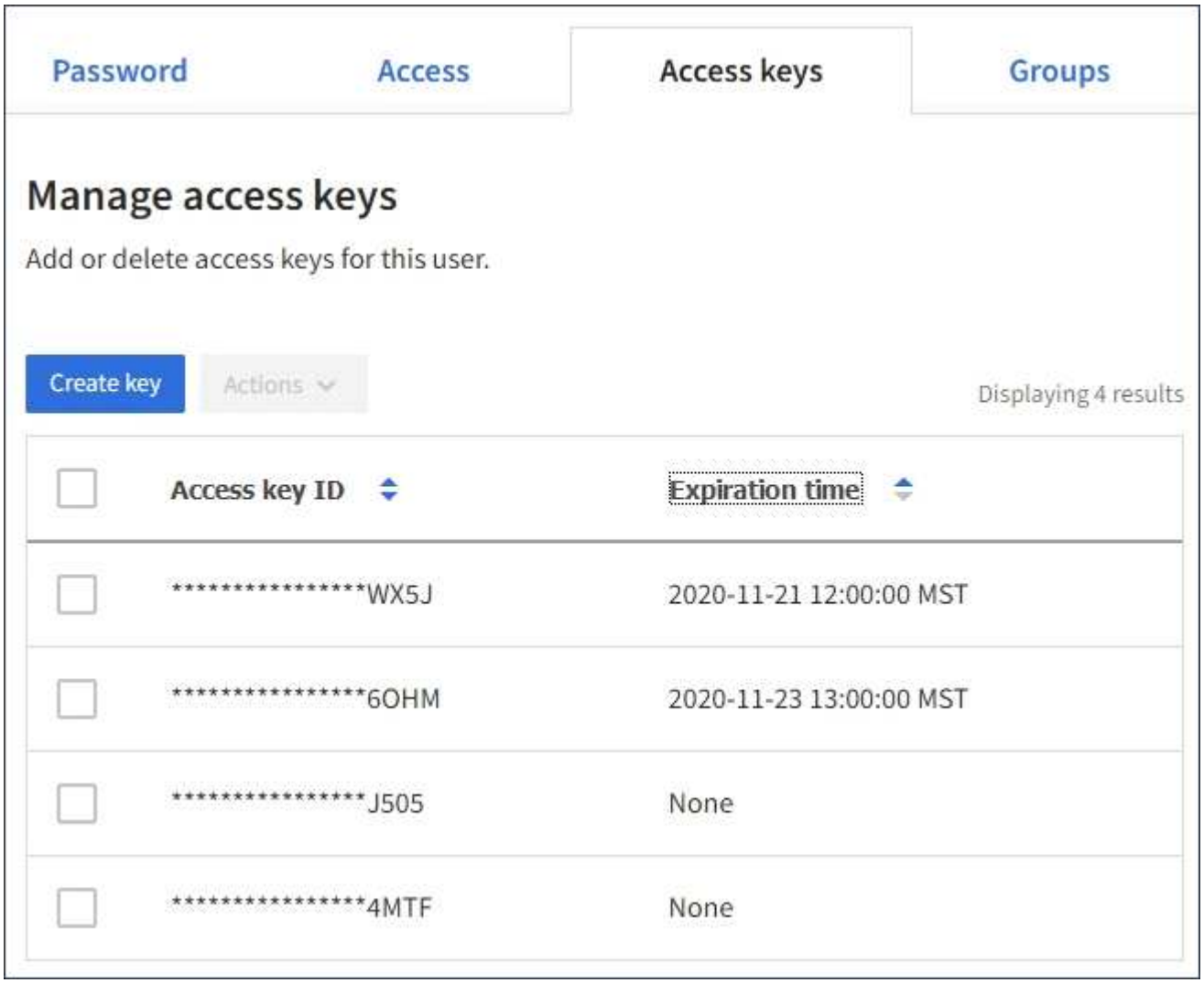
1. Select **ACCESS MANAGEMENT** > **Users**.

The Users page appears and lists the existing users.

2. Select the user whose S3 access keys you want to view.

   The User details page appears.

3. 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.

What you'll need

- You must be signed in to the Tenant Manager using a supported web browser.
- You must have the Root Access permission. See Tenant management permissions.

> 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**.

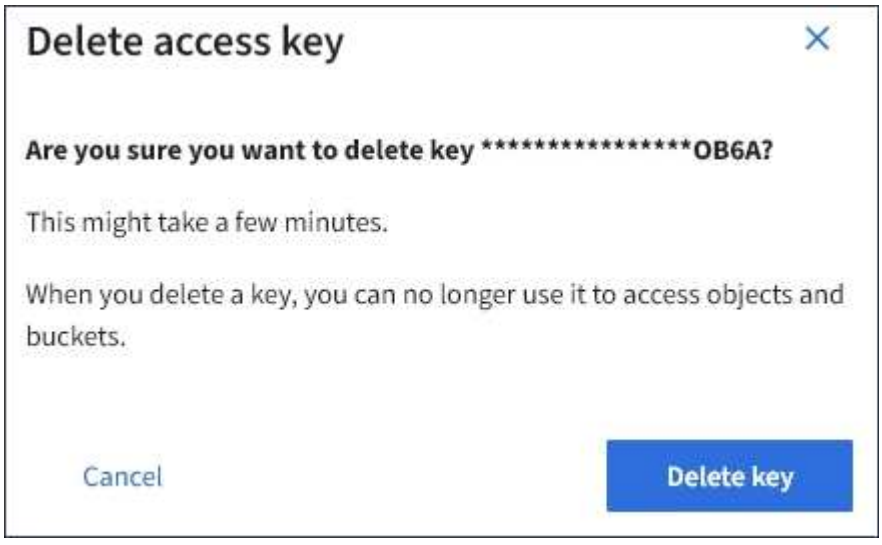
   The Users page appears and lists the existing users.

2. Select the user whose S3 access keys you want to manage.

   The User details page appears.

3. Select **Access keys**, and then select the check box for each access key you want to delete.
4. Select **Actions** > **Delete selected key**.

   A confirmation dialog box appears.



5. Select **Delete key**.

   A confirmation message appears in the upper right corner of the page. Changes might take up to 15 minutes to take effect because of caching.

Manage S3 buckets

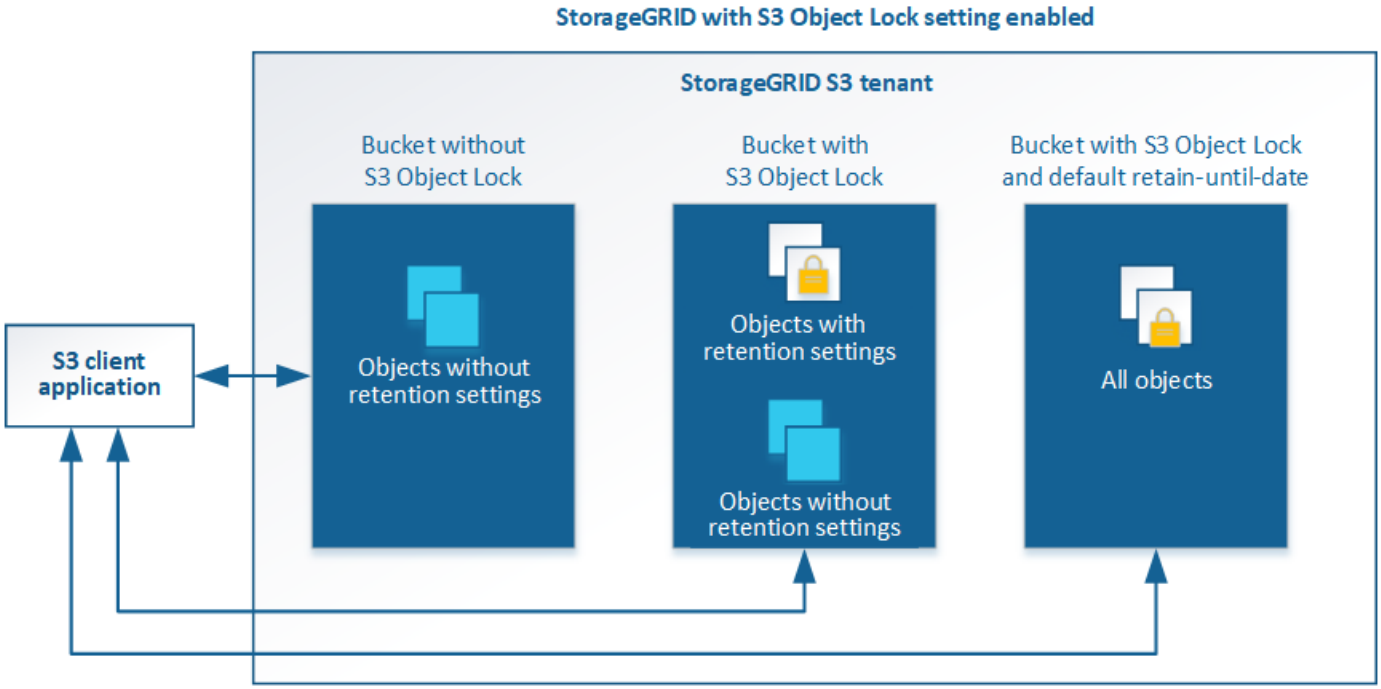
Use S3 Object Lock with tenants

You can use the S3 Object Lock feature in StorageGRID if your objects must comply with regulatory requirements for retention.

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).

As shown in the figure, 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, S3 client applications can optionally specify retention settings for any object version in that bucket. An object version must have retention settings specified to be protected by S3 Object Lock.



The StorageGRID S3 Object Lock feature provides a single retention mode that is equivalent to the Amazon S3 compliance mode. By default, a protected object version cannot be overwritten or deleted by any user. The StorageGRID S3 Object Lock feature does not support a governance mode, and it does not allow users with special permissions to bypass retention settings or to delete protected objects.

If a bucket has S3 Object Lock enabled, the S3 client application can optionally specify either or both of the following object-level retention settings when creating or updating an object:

- **Retain-until-date**: If an object version's retain-until-date is in the future, the object can be retrieved, but it cannot be modified or deleted. As required, an object's retain-until-date can be increased, but this date cannot be decreased.
- **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.

You can also specify a default retention mode and default retention period for the bucket. These are applied to each object added to the bucket that does not specify its own retention settings.

For details on these settings, see Use S3 object lock.

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 the NetApp Knowledge Base article.

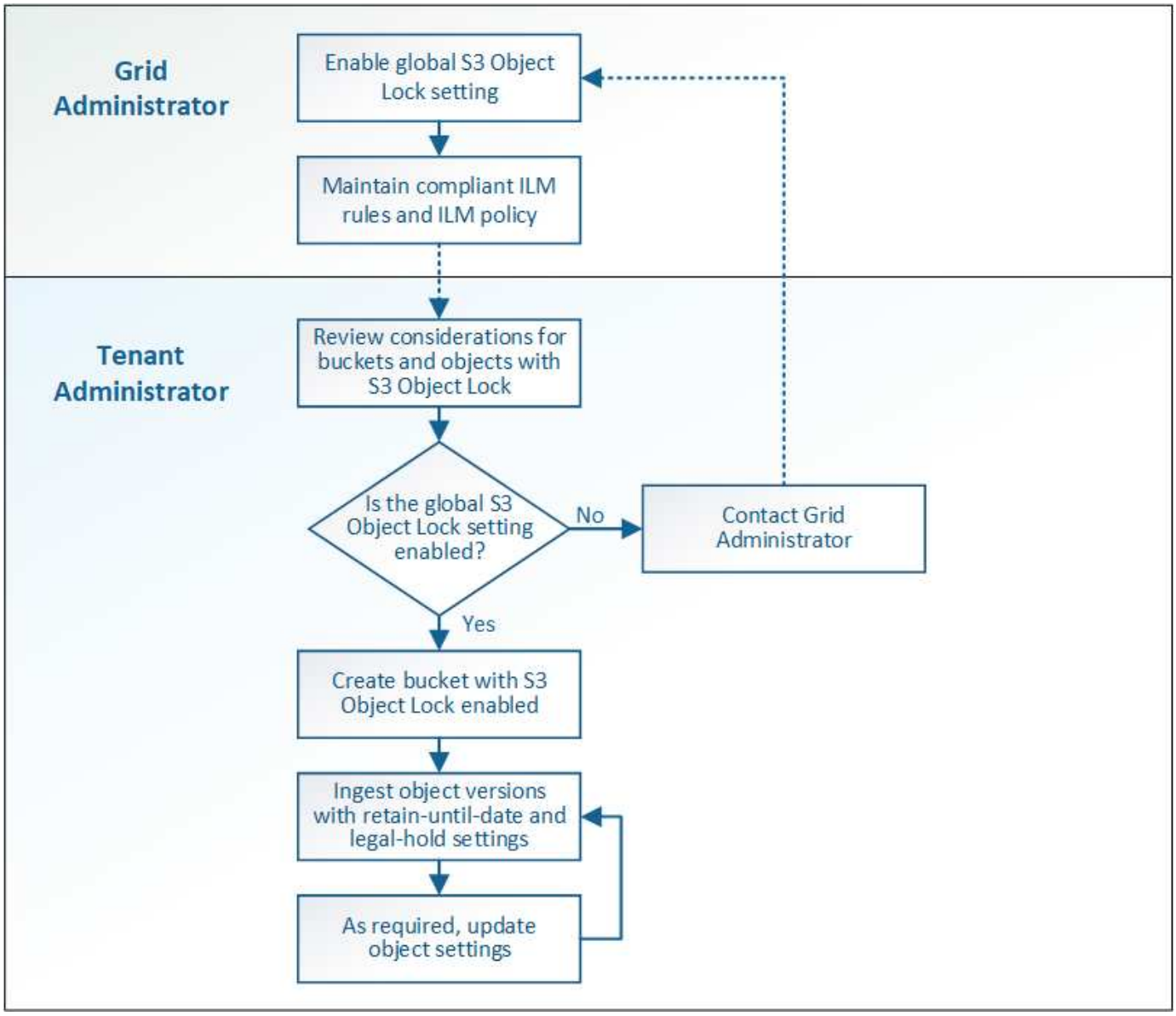
NetApp Knowledge Base: How to manage legacy Compliant buckets in StorageGRID 11.5

S3 Object Lock workflow

The workflow diagram shows the high-level steps for using the S3 Object Lock feature in StorageGRID.

Before you can create buckets with S3 Object Lock enabled, the grid administrator must enable the global S3 Object Lock setting for the entire StorageGRID system. The grid administrator must also ensure that the information lifecycle management (ILM) policy is “compliant”; it must meet the requirements of buckets with S3 Object Lock enabled. For details, contact your grid administrator or see the instructions for managing objects with information lifecycle management.

After the global S3 Object Lock setting has been enabled, you can create buckets with S3 Object Lock enabled. You can then use the S3 client application to optionally specify retention settings for each object version.



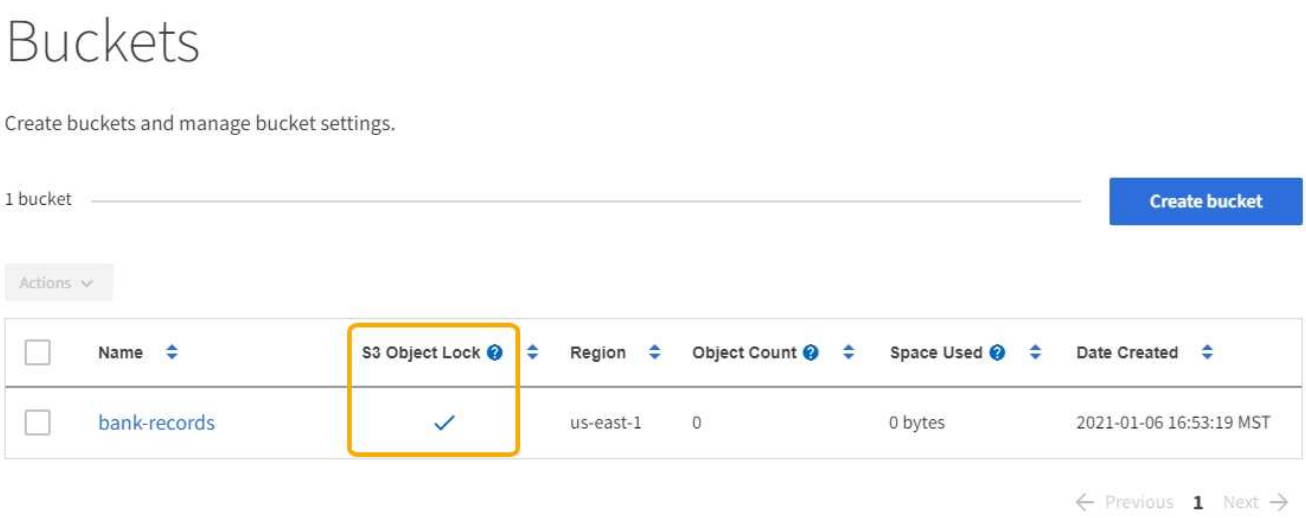
Requirements for S3 Object Lock

Before enabling S3 Object Lock for a bucket, review the requirements for S3 Object Lock buckets and objects and the lifecycle of objects in buckets with S3 Object Lock enabled.

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.

  This example from the Tenant Manager shows a bucket 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 cannot enable S3 Object Lock for an existing bucket.
- Bucket versioning is required with S3 Object Lock. When S3 Object Lock is enabled for a bucket, StorageGRID automatically enables versioning for that bucket.
- After you create a bucket with S3 Object Lock enabled, you cannot disable S3 Object Lock or suspend versioning for that bucket.
- Optionally, you can configure default retention for a bucket. When an object version is uploaded, the default retention is applied to the object version. You can override the bucket default by specifying a retention mode and retain-until-date in the request to upload an object version.
- Bucket lifecycle configuration is supported for S3 Object Lifecycle buckets.
- 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, the S3 client application must either configure bucket default retention, or specify retention settings in each upload request.
- You can increase the retain-until-date for an object version, but you can never decrease this value.
- If you are notified of a pending legal action or regulatory investigation, you can preserve relevant information by placing a legal hold on an object version. When an object version is under a legal hold, that object cannot be deleted from StorageGRID, even if it has reached its retain-until-date. As soon as the legal hold is lifted,the object version can be deleted if the retain-until-date has been reached.
- S3 Object Lock requires the use of versioned buckets. 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 three stages:

1. **Object ingest**
   - When adding an object version to a bucket with S3 Object Lock enabled, the S3 client application can optionally specify retention settings for the object (retain-until-date, legal hold, or both). StorageGRID then generates metadata for that object, which includes a unique object identifier (UUID) and the ingest

date and time.

- After an object version with retention settings is ingested, its data and S3 user-defined metadata cannot be modified.
- StorageGRID stores the object metadata independently of the object data. It maintains three copies of all object metadata at each site.

2. **Object retention**
   - Multiple copies of the object are stored by StorageGRID. The exact number and type of copies and the storage locations are determined by the compliant rules in the active ILM policy.
3. **Object deletion**
   - An object can be deleted when its retain-until-date is reached.
   - An object that is under a legal hold cannot be deleted.

Create S3 bucket

You can use the Tenant Manager to create S3 buckets for object data. When you create a bucket, you must specify the bucket's name and region. If the global S3 Object Lock setting is enabled for the StorageGRID system, you can optionally enable S3 Object Lock for the bucket.

What you'll need

- 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 the Root Access permission. These permissions override the permissions settings in group or bucket policies.

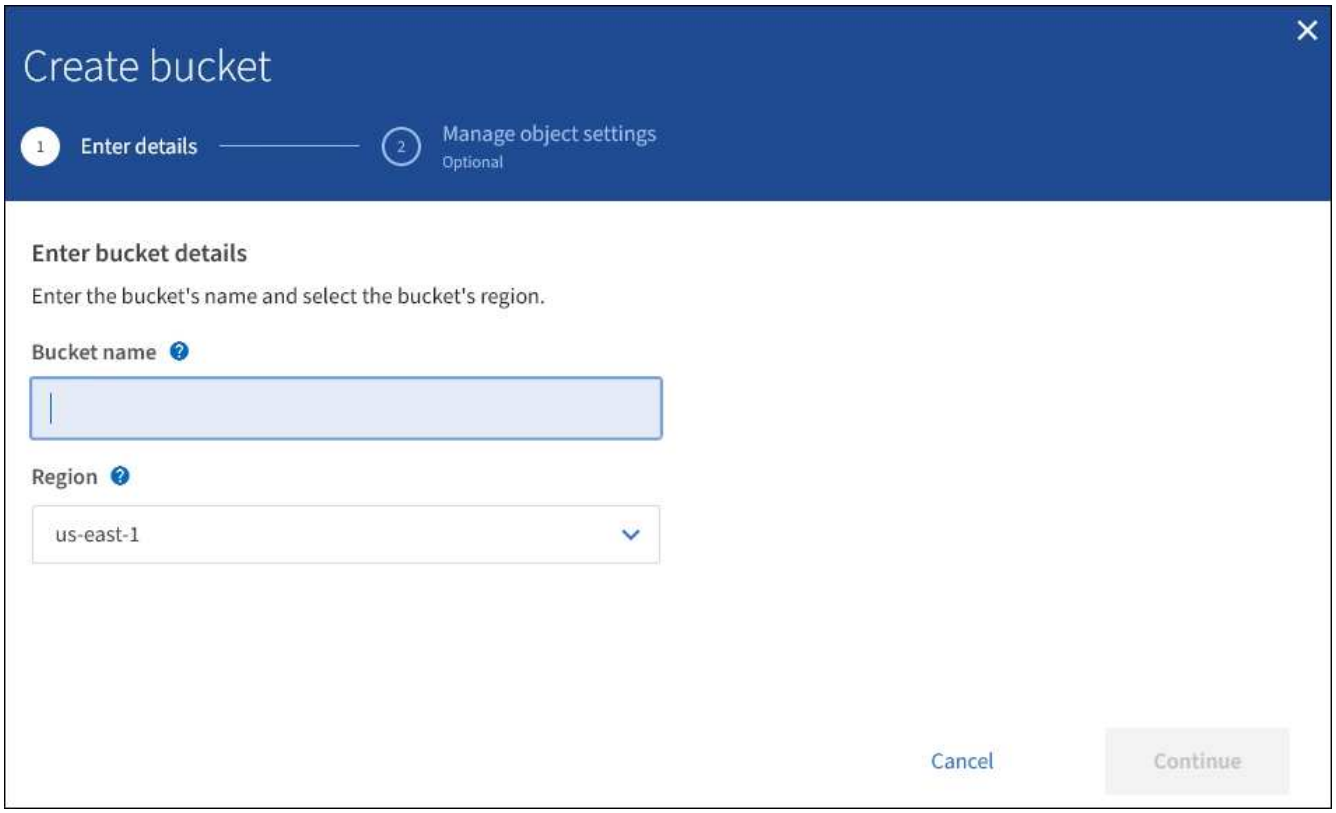
  Permissions to set or modify S3 Object Lock properties of buckets or objects can be granted by bucket policy or group policy.

- If you plan to create a bucket with S3 Object Lock, you have enabled the global S3 Object Lock setting for the StorageGRID system, and you have reviewed the requirements for S3 Object Lock buckets and objects.

  Use S3 Object Lock

Steps

1. Select **STORAGE (S3)** > **Buckets**.
2. Select **Create bucket**.



3. Enter a unique name for the bucket.

   You cannot change the bucket name after creating the bucket.

   Bucket names must comply 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.
   - Should not use 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.

4. Select the region for this bucket.

   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.

   You cannot change the region after creating the bucket.

5. Select **Continue**.
6. 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.

7. If the S3 Object Lock section appears, optionally enable S3 Object Lock for the bucket.

   > ⓘ You cannot enable or disable S3 Object Lock after creating the bucket.

   The S3 Object Lock section appears only if the global S3 Object Lock setting is enabled.

   S3 Object Lock must be enabled for the bucket before an S3 client application can specify retain-until-date and legal hold settings for the objects added to the bucket.

   If you enable S3 Object Lock for a bucket, bucket versioning is enabled automatically. You can also specify a default retention mode and default retention period for the bucket that are applied to each object ingested to the bucket that does not specify its own retention settings.

8. Select **Create bucket**.

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

Related information

Manage objects with ILM

Understand Tenant Management API

Use S3

View S3 bucket details

You can view a list of the buckets and bucket settings in your tenant account.

What you'll need

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

Steps

1. Select **STORAGE (S3)** > **Buckets**.

   The Buckets page appears and lists all buckets for the tenant account.

2. Review the information for each bucket.

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

   - Name: The bucket's unique name, which cannot be changed.
   - S3 Object Lock: Whether S3 Object Lock is enabled for this bucket.

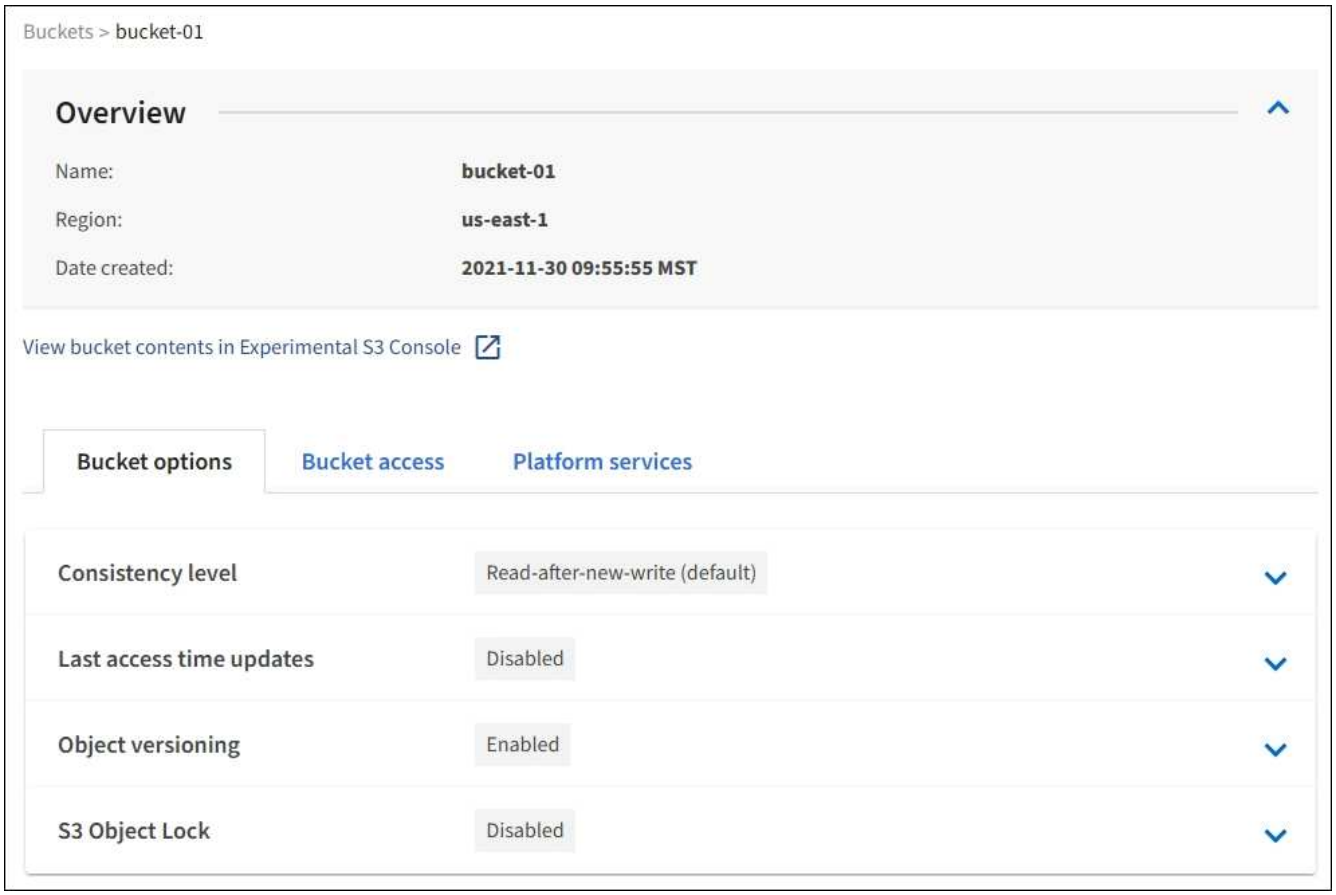
     This column is not displayed if the global S3 Object lock setting is disabled. This column also shows information for any legacy Compliant buckets.

   - Region: The bucket's region, which cannot be changed.
   - Object Count: The number of objects in this bucket.
   - Space Used: The logical size of all objects in this bucket. The logical size does not include the actual space required for replicated or erasure-coded copies or for object metadata.
   - Date Created: The date and time the bucket was created.

   > 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. To view and manage the settings for a bucket, select the bucket name.

   The bucket details page allows you to view and edit the settings for bucket options, bucket access, and platform services.



Change the consistency level

If you are using an S3 tenant, you can use the Tenant Manager or the Tenant Management API to change the consistency control for operations performed on the objects in S3 buckets.

What you'll need

- You must be signed in to the Tenant Manager using a supported web browser.
- You must belong to a user group that has the Manage All Buckets or the Root Access permission. These permissions override the permissions settings in group or bucket policies. See Tenant management permissions.

About this task

Consistency level provides a balance between the availability of the objects and the consistency of those objects across different Storage Nodes and sites. In general, you should use the **Read-after-new-write** consistency level for your buckets.

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

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

Steps

1. Select **STORAGE (S3)** > **Buckets**.
2. Select the bucket name from the list.

   The bucket details page appears.

3. Select **Bucket options** > **Consistency level**.
4. Select a consistency level 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 do not exist). Not supported for S3 FabricPool buckets.
5. Select **Save changes**.

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 in its placement instructions. You can ignore these instructions if your StorageGRID system does not include such a rule.

What you'll need

- You must be signed in to the Tenant Manager using a supported web browser.
- You must belong to a user group that has the Manage All Buckets or the Root Access permission. These permissions override the permissions settings in group or bucket policies. See Tenant management permissions.

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.

> See the instructions for managing objects with information lifecycle management.

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.

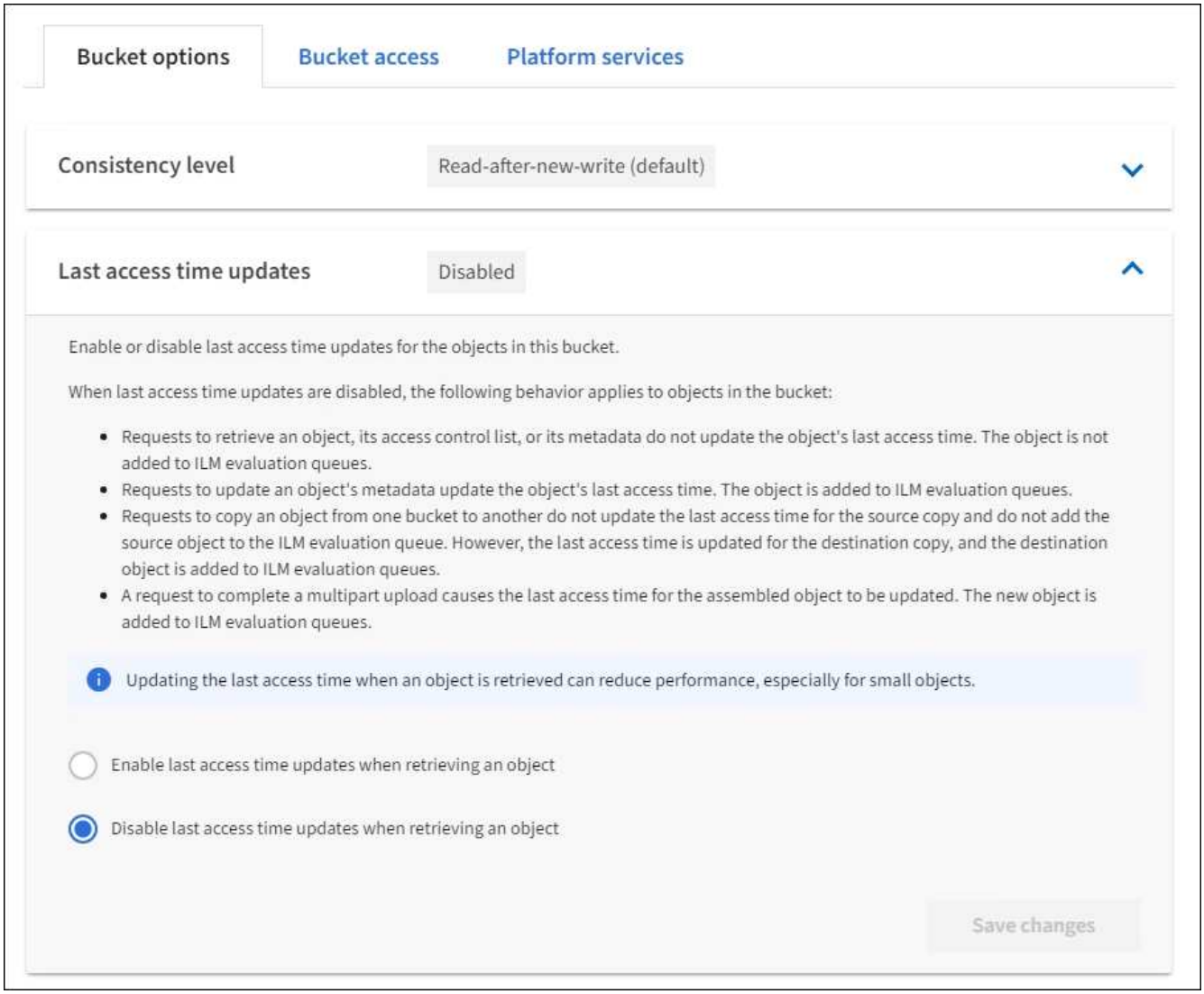
Type of request	Behavior if last access time is disabled (default)		Behavior if last access time is enabled	
	Last access time updated?	Object added to ILM evaluation queue?	Last access time updated?	Object added to ILM evaluation queue?
Request to retrieve an object, its access control list, or its metadata	No	No	Yes	Yes
Request to update an object's metadata	Yes	Yes	Yes	Yes
Request to 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 **STORAGE (S3)** > **Buckets**.
2. Select the bucket name from the list.

   The bucket details page appears.

3. Select **Bucket options** > **Last access time updates**.
4. Select the appropriate radio button to enable or disable last access time updates.



5. Select **Save changes**.

Related information

Tenant management permissions

Manage objects with ILM

Change object versioning for a bucket

If you are using an S3 tenant, you can use the Tenant Manager or the Tenant Management API to change the versioning state for S3 buckets.

What you'll need

- 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 the Root Access permission. These permissions override the permissions settings in group or bucket policies.

  Tenant management permissions

About this task

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

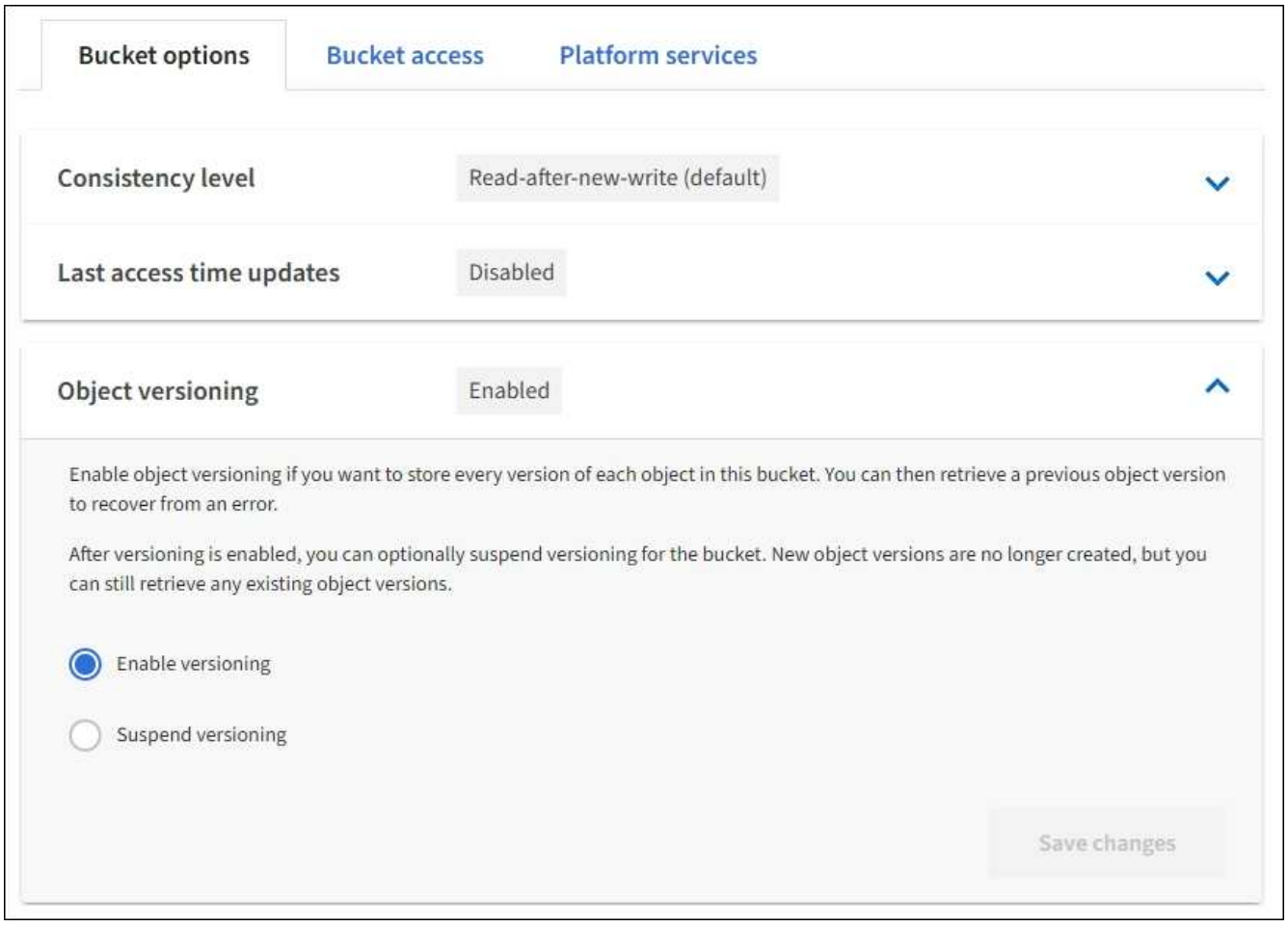
- Disabled: Versioning has never been enabled
- Enabled: Versioning is enabled
- Suspended: Versioning was previously enabled and is suspended

S3 object versioning

ILM rules and policies for S3 versioned objects (Example 4)

Steps

1. Select **STORAGE (S3)** > **Buckets**.
2. Select the bucket name from the list.
3. Select **Bucket options** > **Object versioning**.



4. Select a versioning state for the objects in this bucket.

> 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**.

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.

What you'll need

- You must be signed in to the Tenant Manager using a supported web browser.
- You must belong to a user group that has the Manage All Buckets or the Root Access permission. These permissions override the permissions settings in group or bucket policies.

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`.

Steps

1. Use a text editor to create the XML required to enable CORS.

   This example shows the XML used to enable CORS for an S3 bucket. This XML allows any domain to send GET requests to the bucket, but it only allows the `http://www.example.com` domain to send POST and DELETE requests. All request headers are allowed.

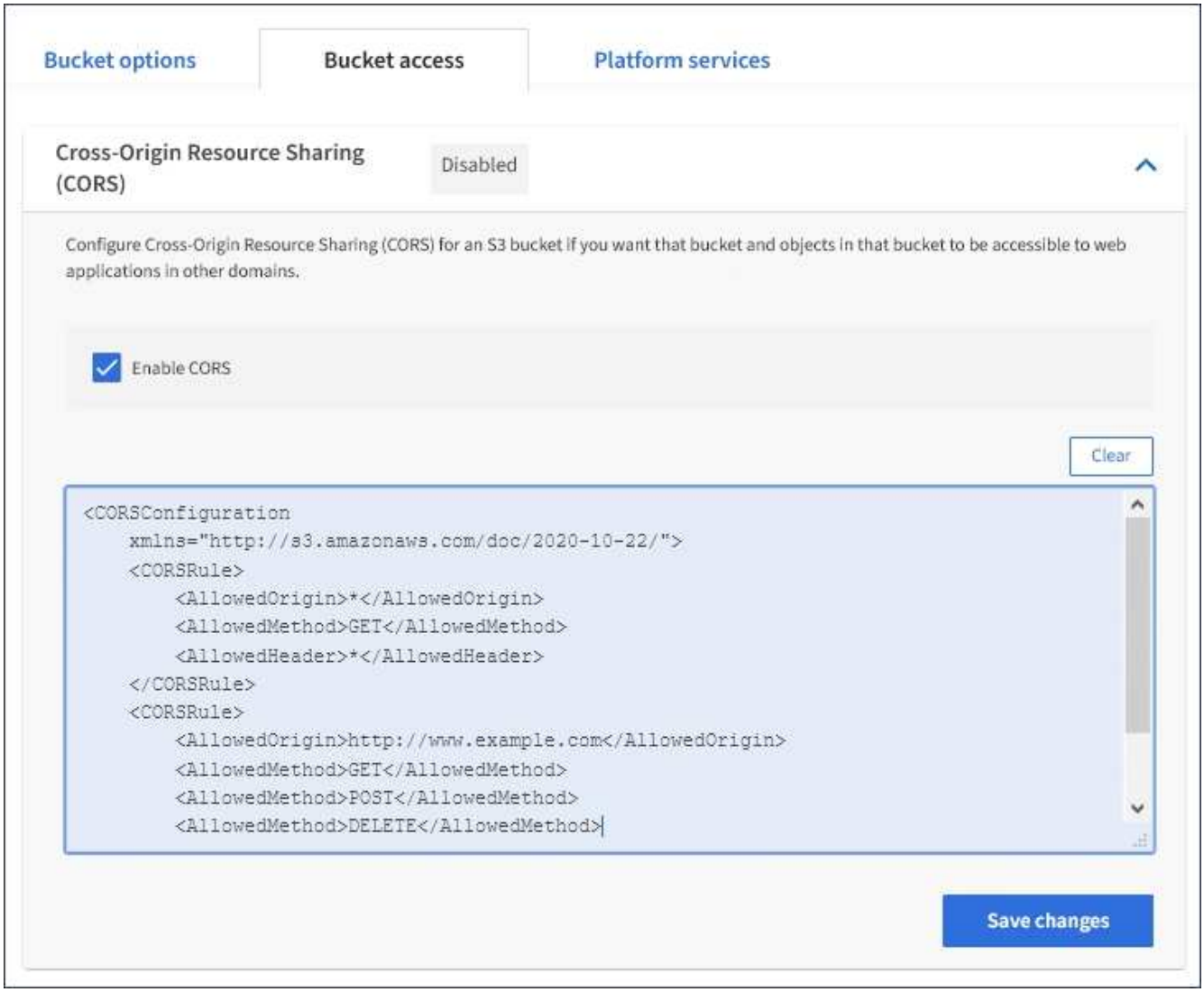
```
<CORSConfiguration
    xmlns="http://s3.amazonaws.com/doc/2020-10-22/">
    <CORSRule>
        <AllowedOrigin>*</AllowedOrigin>
        <AllowedMethod>GET</AllowedMethod>
        <AllowedHeader>*</AllowedHeader>
    </CORSRule>
    <CORSRule>
        <AllowedOrigin>http://www.example.com</AllowedOrigin>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

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

   The bucket details page appears.

4. Select **Bucket access** > **Cross-Origin Resource Sharing (CORS)**.
5. Select the **Enable CORS** check box.
6. Paste the CORS configuration XML into the text box, and select **Save changes**.



7. To modify the CORS setting for the bucket, update the CORS configuration XML in the text box or select **Clear** to start over. Then select **Save changes**.
8. To disable CORS for the bucket, unselect the **Enable CORS** check box, and then select **Save changes**.

Delete S3 bucket

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

What you'll need

- You must be signed in to the Tenant Manager using a supported web browser.
- You must belong to a user group that has the Manage All Buckets or the Root Access permission. These permissions override the permissions settings in group or bucket policies. See Tenant management permissions.
- The buckets you want to delete are empty.

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 cannot delete an S3 bucket if it contains objects or noncurrent object versions. For information about how

S3 versioned objects are deleted, see the instructions for managing objects with information lifecycle management.

Steps

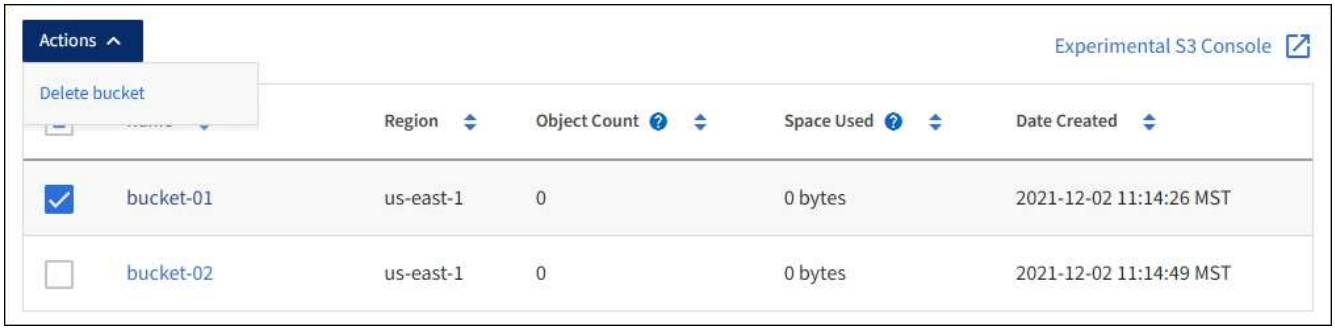
1. Select **STORAGE (S3)** > **Buckets**.

   The Buckets page appears and shows all existing S3 buckets.

2. Select the check box for the empty bucket you want to delete. You can select more than one bucket at a time.

   The Actions menu is enabled.

3. From the Actions menu, select **Delete bucket** (or **Delete buckets** if you have chosen more than one).



4. When the confirmation dialog box appears, select **Yes** to delete all of the buckets you have chosen.

   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 before you can delete a bucket.

Use Experimental S3 Console

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

You can also use S3 Console to do the following:

- Add and delete objects, object versions, and folders
- Rename objects
- Move and copy objects between buckets and folders
- Manage object tags
- View object metadata
- Download objects

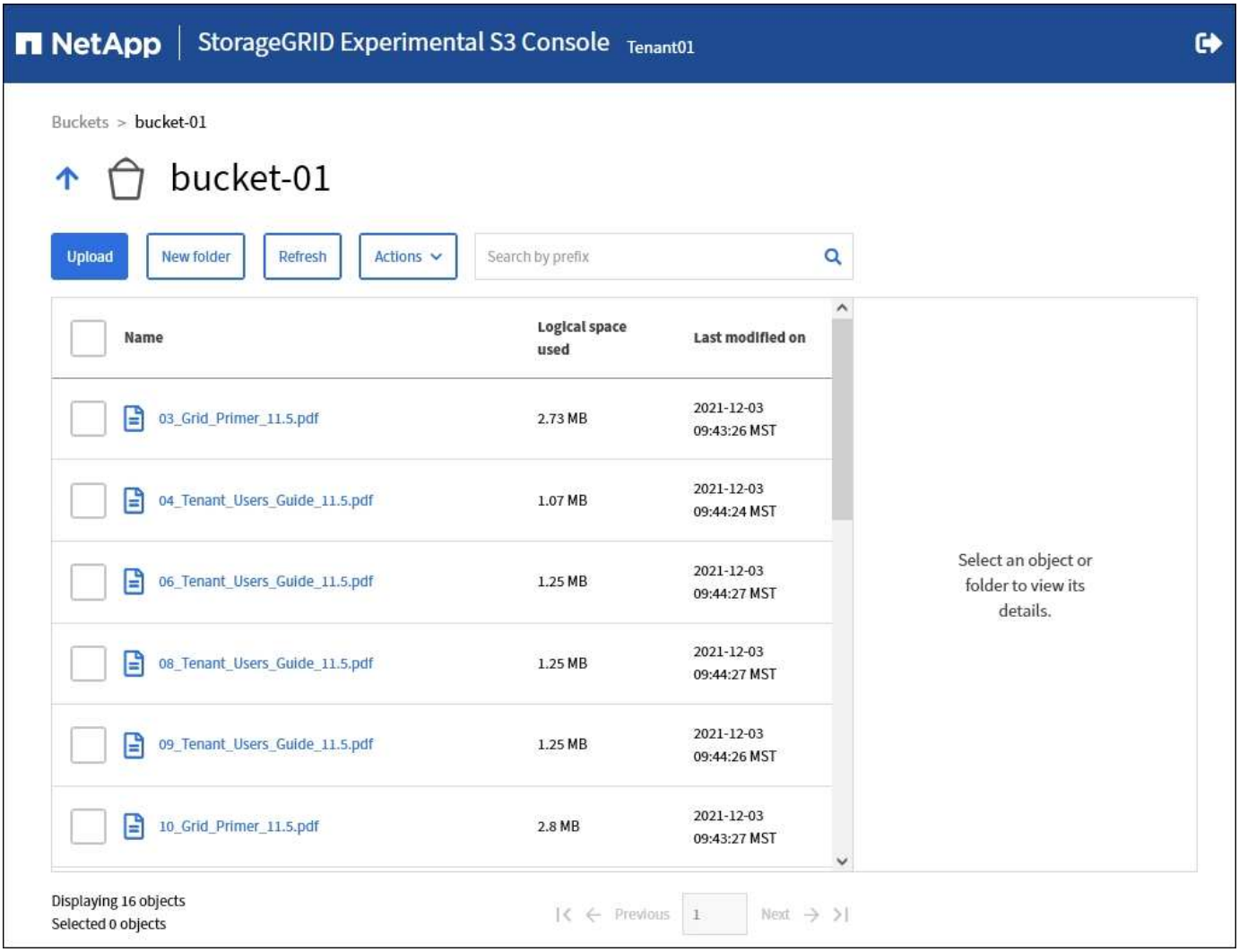
> S3 Console has not been fully tested and is marked as “experimental.” It is not intended for bulk management of objects or for use in a production environment. Tenants should only use S3 Console when performing functions for a small number of objects, such as when uploading objects to simulate a new ILM policy, troubleshooting ingest issues, or using proof-of-concept or non-production grids.

What you’ll need

- You are signed in to the Tenant Manager using a supported web browser.
- You have the Manage Your Own S3 Credentials permission.
- You have created a bucket.
- 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 **Buckets**.
2. Select Experimental S3 Console . You can also access this link from the bucket details page.
3. On the Experimental S3 Console sign-in page, 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. Manage objects as needed.


NetApp | StorageGRID Experimental S3 Console Tenant01
Buckets > bucket-01
bucket-01
Upload
New folder
Refresh
Actions
Search by prefix
Name
Logical space used
Last modified on
03_Grid_Primer_11.5.pdf
2.73 MB
2021-12-03 09:43:26 MST
04_Tenant_Users_Guide_11.5.pdf
1.07 MB
2021-12-03 09:44:24 MST
06_Tenant_Users_Guide_11.5.pdf
1.25 MB
2021-12-03 09:44:27 MST
08_Tenant_Users_Guide_11.5.pdf
1.25 MB
2021-12-03 09:44:27 MST
09_Tenant_Users_Guide_11.5.pdf
1.25 MB
2021-12-03 09:44:26 MST
10_Grid_Primer_11.5.pdf
2.8 MB
2021-12-03 09:43:27 MST
Select an object or folder to view its details.
Displaying 16 objects
Selected 0 objects
Previous
1
Next

Copyright information

Trademark information