



S3 object storage management

ONTAP 9

NetApp
April 24, 2024

Table of Contents

- S3 object storage management 1
 - Learn about S3 support in ONTAP 9 1
 - Plan 4
 - Configure 8
 - Protect buckets with S3 SnapMirror 55
 - Audit S3 events 90

S3 object storage management

Learn about S3 support in ONTAP 9

S3 configuration overview

Beginning with ONTAP 9.8, you can enable an ONTAP Simple Storage Service (S3) object storage server in an ONTAP cluster.

ONTAP supports two on-premises use case scenarios for serving S3 object storage:

- FabricPool tier to a bucket on local cluster (tier to a local bucket) or remote cluster (cloud tier).
- S3 client app access to a bucket on the local cluster or a remote cluster.

Beginning with ONTAP 9.14.1, you can enable an S3 object storage server on an SVM in a mirrored or unmirrored aggregate in MetroCluster IP and FC configurations.

Beginning with ONTAP 9.12.1, you can enable an S3 object storage server on an SVM in an unmirrored aggregate in a MetroCluster IP configuration. For more information on the limitations of unmirrored aggregates in MetroCluster IP configurations, see [Considerations for unmirrored aggregates](#).

You should use these procedures if you want to configure S3 object storage in the following way:

- You want to provide S3 object storage from an existing cluster running ONTAP.

ONTAP S3 is appropriate if you want S3 capabilities on existing clusters without additional hardware and management. However, NetApp StorageGRID software continues to be the NetApp flagship solution for object storage. For more information, see the [StorageGRID documentation](#).

- You have cluster administrator privileges, not SVM administrator privileges.

S3 configuration with System Manager and the ONTAP CLI

You can configure and manage ONTAP S3 with System Manager and the ONTAP CLI. When you enable S3 and create buckets using System Manager, ONTAP selects best-practice defaults for simplified configuration. If you need to specify configuration parameters, you might want to use the ONTAP CLI. If you configure the S3 server and buckets from the CLI, you can still manage them with System Manager if desired, or vice-versa.

When you create an S3 bucket using System Manager, ONTAP configures a default performance service level that is the highest available on your system. For example, on an AFF system, the default setting would be **Extreme**. Performance service levels are predefined adaptive Quality of Service (QoS) policy groups. Instead of one of the default service levels, you can specify a custom QoS policy group or no policy group.

Predefined adaptive QoS policy groups are:

- **Extreme**: Used for applications that expect the lowest latency and highest performance.
- **Performance**: Used for applications with modest performance needs and latency.
- **Value**: Used for applications for which throughput and capacity are more important than latency.
- **Custom**: Specify a custom QoS policy or no QoS policy.

If you select **Use for tiering**, no performance service levels are selected, and the system tries to select low-

cost media with optimal performance for the tiered data.

See also: [Use adaptive QoS policy groups](#).

ONTAP tries to provision this bucket on local tiers that have the most appropriate disks, satisfying the chosen service level. However, if you need to specify which disks to include in the bucket, consider configuring S3 object storage from the CLI by specifying the local tiers (aggregate). If you configure the S3 server from the CLI, you can still manage it with System Manager if desired.

If you want the ability to specify which aggregates are used for buckets, you can only do so using the CLI.

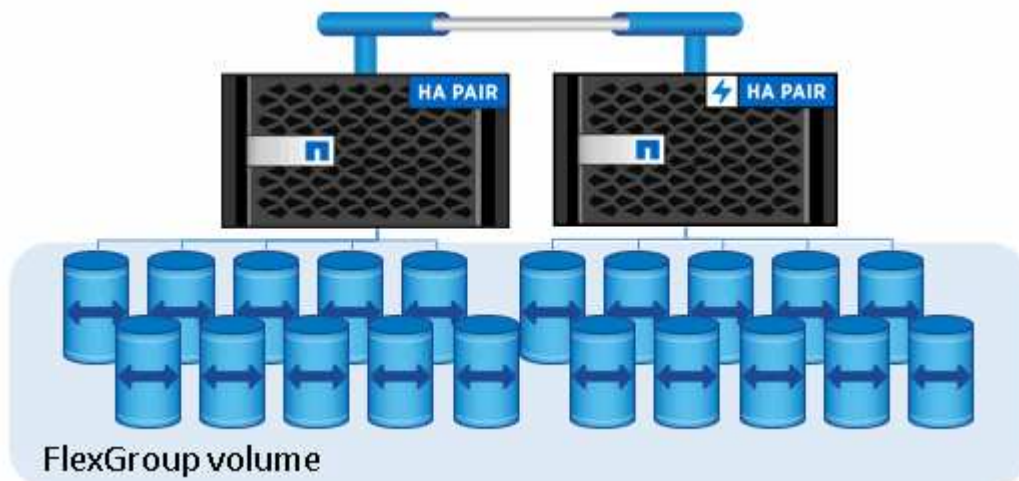
Configuring S3 buckets on Cloud Volumes ONTAP

If you want to serve buckets from Cloud Volumes ONTAP, it is strongly recommended that you manually select the underlying aggregates to ensure that they are using one node only. Using aggregates from both nodes can impact performance, because the nodes will be in geographically separated availability zones and hence susceptible to latency issues. Therefore, in Cloud Volumes ONTAP environments, you should [configure S3 buckets from the CLI](#).

Otherwise, S3 servers on Cloud Volumes ONTAP are configured and maintained the same in Cloud Volumes ONTAP as in on-premises environments.

Architecture

In ONTAP, the underlying architecture for a bucket is a FlexGroup volume—a single namespace that is made up of multiple constituent member volumes but is managed as a single volume.



Buckets are only limited by the physical maximums of the underlying hardware, architectural maximums could be higher. Buckets can take advantage of FlexGroup elastic sizing to automatically grow a constituent of a FlexGroup volume if it is running out of space. There is a limit of 1000 buckets per FlexGroup volume, or 1/3 of the FlexGroup volume's capacity (to account for data growth in buckets).



No NAS or SAN protocol access is permitted to the FlexGroup volume that contain S3 buckets.

Access to the bucket is provided through authorized users and client applications.



Use cases

There are three primary use cases for client access to ONTAP S3 services:

- For ONTAP systems using ONTAP S3 as a remote FabricPool capacity (cloud) tier

The S3 server and bucket containing the capacity tier (for *cold* data) is on a different cluster than the performance tier (for *hot* data).

- For ONTAP systems using ONTAP S3 as a local FabricPool tier

The S3 server and bucket containing the capacity tier is on the same cluster, but on a different HA pair, as the performance tier.

- For external S3 client apps

ONTAP S3 serves S3 client apps run on non-NetApp systems.

It is a best practice to provide access to ONTAP S3 buckets using HTTPS. When HTTPS is enabled, security certificates are required for proper integration with SSL/TLS. Client users' access and secret keys are then required to authenticate the user with ONTAP S3 as well as authorizing the users' access permissions for operations within ONTAP S3. The client application should also have access to the root CA certificate (the ONTAP S3 server's signed certificate) to be able to authenticate the server and create a secure connection between client and server.

Users are created within the S3-enabled SVM, and their access permissions can be controlled at the bucket or SVM level; that is, they can be granted access to one or more buckets within the SVM.

HTTPS is enabled by default on ONTAP S3 servers. It is possible to disable HTTPS and enable HTTP for client access, in which case authentication using CA certificates is not required. However, when HTTP is enabled and HTTPS is disabled, all communication with the ONTAP S3 server are sent over the network in clear text.

For additional information, see [Technical Report: S3 in ONTAP Best Practices](#)

Related information

[FlexGroup volumes management](#)

Plan

ONTAP version support for S3 object storage

ONTAP supports S3 object storage for on-premises environments beginning with ONTAP 9.8. Cloud Volumes ONTAP supports S3 object storage for cloud environments beginning with ONTAP 9.9.1.

S3 support with Cloud Volumes ONTAP

ONTAP S3 is configured and functions the same in Cloud Volumes ONTAP as in on-premises environments, with one exception:

- Underlying aggregates should be from one node only. Learn more about [bucket creation in CVO environments](#).

Cloud Provider	ONTAP Version
Azure	ONTAP 9.9.1 and later
AWS	ONTAP 9.11.0 and later
Google Cloud	ONTAP 9.12.1 and later

S3 public preview in ONTAP 9.7

In ONTAP 9.7, S3 object storage was introduced as a public preview. That version was not intended for production environments and will no longer be updated as of ONTAP 9.8. Only ONTAP 9.8 and later releases support S3 object storage in production environments.

S3 buckets created with the 9.7 public preview can be used in ONTAP 9.8 and later, but cannot take advantage of feature enhancements. If you have buckets created with the 9.7 public preview, you should migrate the contents of those buckets to 9.8 buckets for feature support, security, and performance enhancements.

ONTAP S3 supported actions

ONTAP S3 actions are supported by standard S3 REST APIs except as indicated below. For details, see the [Amazon S3 API Reference](#).

Bucket operations

The following operations are supported in ONTAP using AWS S3 APIs:

Bucket operation	ONTAP support beginning with
CreateBucket	ONTAP 9.11.1
DeleteBucket	ONTAP 9.11.1

Bucket operation	ONTAP support beginning with
DeleteBucketPolicy	ONTAP 9.12.1
GetBucketAcl	ONTAP 9.8
GetBucketLifecycleConfiguration	ONTAP 9.13.1 * only expiration actions are supported
GetBucketLocation	ONTAP 9.10.1
GetBucketPolicy	ONTAP 9.12.1
HeadBucket	ONTAP 9.8
ListBuckets	ONTAP 9.8
ListBucketVersioning	ONTAP 9.11.1
ListObjectVersions	ONTAP 9.11.1
PutBucket	<ul style="list-style-type: none"> • ONTAP 9.11.1 • ONTAP 9.8 - supported with ONTAP REST APIs only
PutBucketLifecycleConfiguration	ONTAP 9.13.1 * only expiration actions are supported
PutBucketPolicy	ONTAP 9.12.1

Object operations

Beginning with ONTAP 9.9.1, ONTAP S3 supports object metadata and tagging.

- PutObject and CreateMultipartUpload include key-value pairs using `x-amz-meta-<key>`.

For example: `x-amz-meta-project: ontap_s3`.

- GetObject. and HeadObject return user-defined metadata.
- Unlike metadata, tags can be read independently of objects using:
 - PutObjectTagging
 - GetObjectTagging
 - DeleteObjectTagging

Beginning with ONTAP 9.11.1, ONTAP S3 supports object versioning and associated actions with these ONTAP APIs:

- GetBucketVersioning
- ListBucketVersions
- PutBucketVersioning

Object operation	ONTAP support beginning with
AbortMultipartUpload	ONTAP 9.8

Object operation	ONTAP support beginning with
CompleteMultipartUpload	ONTAP 9.8
CopyObject	ONTAP 9.12.1
CreateMultipartUpload	ONTAP 9.8
DeleteObject	ONTAP 9.8
DeleteObjects	ONTAP 9.11.1
DeleteObjectTagging	ONTAP 9.9.1
GetBucketVersioning	ONTAP 9.11.1
GetObject	ONTAP 9.8
GetObjectAcl	ONTAP 9.8
GetObjectRetention	ONTAP 9.14.1
GetObjectTagging	ONTAP 9.9.1
HeadObject	ONTAP 9.8
ListMultipartUpload	ONTAP 9.8
ListObjects	ONTAP 9.8
ListObjectsV2	ONTAP 9.8
ListBucketVersions	ONTAP 9.11.1
ListParts	ONTAP 9.8
PutBucketVersioning	ONTAP 9.11.1
PutObject	ONTAP 9.8
PutObjectLockConfiguration	ONTAP 9.14.1
PutObjectRetention	ONTAP 9.14.1
PutObjectTagging	ONTAP 9.9.1
UploadPart	ONTAP 9.8
UploadPartCopy	ONTAP 9.12.1

Group policies

These operations are not specific to S3 and are generally associated with Identity and Management (IAM) processes. ONTAP supports these commands but does not use the IAM REST APIs.

- Create Policy
- AttachGroup Policy

User management

These operations are not specific to S3 and are generally associated with IAM processes.

- CreateUser

- DeleteUser
- CreateGroup
- DeleteGroup

ONTAP S3 interoperability

The ONTAP S3 server interacts normally with other ONTAP functionality except as noted in this table.

Feature area	Supported	Not supported
Cloud Volumes ONTAP	<ul style="list-style-type: none"> • Azure clients in ONTAP 9.9.1 and later releases • AWS clients in ONTAP 9.11.0 and later releases • Google Cloud clients in ONTAP 9.12.1 and later releases 	<ul style="list-style-type: none"> • Cloud Volumes ONTAP for any client in ONTAP 9.8 and earlier releases
Data protection	<ul style="list-style-type: none"> • Cloud Sync • Object versioning (beginning with ONTAP 9.11.1) • S3 SnapMirror (beginning with ONTAP 9.10.1) • MetroCluster IP configurations (beginning with ONTAP 9.12.1) • SnapLock (beginning with ONTAP 9.14.1) • WORM (beginning with ONTAP 9.14.1) 	<ul style="list-style-type: none"> • Erasure coding • Information lifecycle management • NDMP • SMTape • SnapMirror Cloud • SVM disaster recovery • SyncMirror • User-created Snapshot copies
Encryption	<ul style="list-style-type: none"> • NetApp Aggregate Encryption (NAE) • NetApp Volume Encryption (NVE) • NetApp Storage Encryption (NSE) • TLS/SSL 	<ul style="list-style-type: none"> • SLAG
Storage efficiency	<ul style="list-style-type: none"> • Deduplication • Compression • Compaction 	<ul style="list-style-type: none"> • Aggregate-level efficiencies • Volume clone of the FlexGroup volume containing ONTAP S3 buckets
Storage virtualization	-	NetApp FlexArray Virtualization

Feature area	Supported	Not supported
Quality of service (QoS)	<ul style="list-style-type: none"> • QoS maximums (ceilings) • QoS minimums (floors) 	-
Additional features	<ul style="list-style-type: none"> • Audit S3 events (beginning with ONTAP 9.10.1) 	<ul style="list-style-type: none"> • FlexCache volumes • FPolicy • Qtrees • Quotas

ONTAP S3 validated third-party solutions

NetApp has validated the following third-party solutions for use with ONTAP S3. If the solution you are looking for is not listed, please contact your NetApp account representative.

Third-party solutions validated on ONTAP S3

NetApp has tested these solutions in collaboration with the respective partners.

- Amazon SageMaker
- Apache Hadoop S3A client
- Apache Kafka
- Commvault (V11)
- Confluent Kafka
- Red Hat Quay
- Rubrik
- Snowflake
- Trino
- Veeam (V12)

Configure

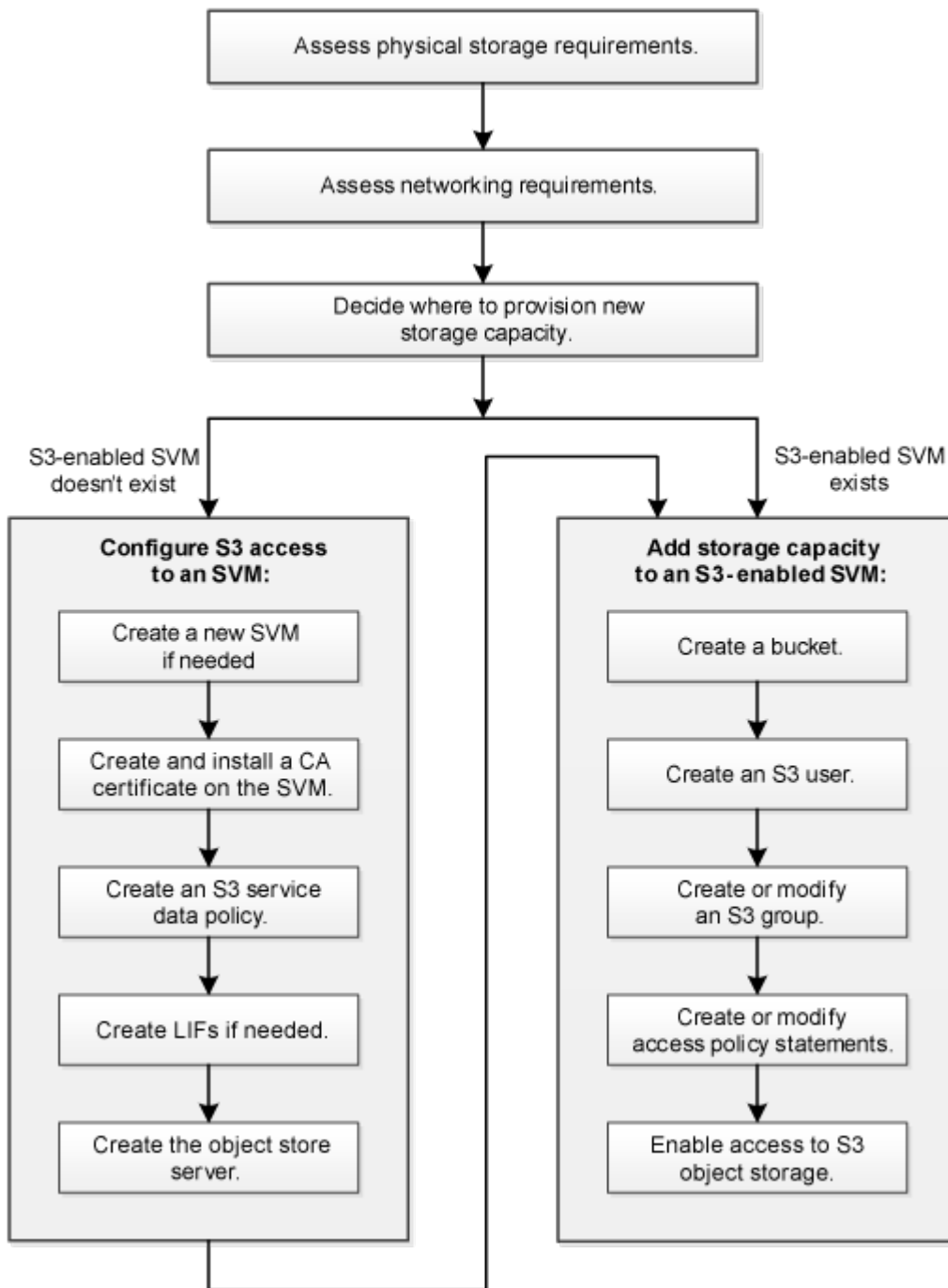
About the S3 configuration process

S3 configuration workflow

Configuring S3 involves assessing physical storage and networking requirements, and then choosing a workflow that is specific to your goal—configuring S3 access to a new or existing SVM, or adding a bucket and users to an existing SVM that is already fully configured for S3 access.

When you configure S3 access to a new storage VM using System Manager, you are prompted to enter certificate and networking information, and the storage VM and S3 object storage server are created in a single

operation.



Assess physical storage requirements

Before provisioning S3 storage for clients, you must ensure that there is sufficient space in existing aggregates for the new object store. If there is not, you can add disks to existing aggregates or create new aggregates of the desired type and location.

About this task

When you create an S3 bucket in an S3-enabled SVM, a FlexGroup volume is automatically created to support the bucket. You can let ONTAP select the underlying aggregates and FlexGroup components automatically (the default) or you can select the underlying aggregates and FlexGroup components yourself.

If you decide to specify the aggregates and FlexGroup components — for example, if you have specific performance requirements for the underlying disks — you should make sure that your aggregate configuration conforms to best practice guidelines for provisioning a FlexGroup volume. Learn more:

- [FlexGroup volumes management](#)
- [NetApp Technical Report 4571-a: NetApp ONTAP FlexGroup Volume Top Best Practices](#)

If you are serving buckets from Cloud Volumes ONTAP, it is strongly recommended that you manually select the underlying aggregates to ensure that they are using one node only. Using aggregates from both nodes can impact performance, because the nodes will be in geographically separated availability zones and hence susceptible to latency issues. Learn about [creating buckets for Cloud Volumes ONTAP](#).

You can use the ONTAP S3 server to create a local FabricPool capacity tier; that is, in the same cluster as the performance tier. This can be useful, for example, if you have SSD disks attached to one HA pair and you want to tier *cold* data to HDD disks in another HA pair. In this use case, the S3 server and the bucket containing the local capacity tier should therefore be in a different HA pair than the performance tier. Local tiering is not supported on one-node and two-node clusters.

Steps

1. Display available space in existing aggregates:

```
storage aggregate show
```

If there is an aggregate with sufficient space or requisite node location, record its name for your S3 configuration.

```
cluster-1::> storage aggregate show
```

Aggregate	Size	Available	Used%	State	#Vols	Nodes	RAID Status
aggr_0	239.0GB	11.13GB	95%	online	1	node1	raid_dp, normal
aggr_1	239.0GB	11.13GB	95%	online	1	node1	raid_dp, normal
aggr_2	239.0GB	11.13GB	95%	online	1	node2	raid_dp, normal
aggr_3	239.0GB	11.13GB	95%	online	1	node2	raid_dp, normal
aggr_4	239.0GB	238.9GB	95%	online	5	node3	raid_dp, normal
aggr_5	239.0GB	239.0GB	95%	online	4	node4	raid_dp, normal

6 entries were displayed.

2. If there are no aggregates with sufficient space or requisite node location, add disks to an existing aggregate by using the `storage aggregate add-disks` command, or create a new aggregate by using the `storage aggregate create` command.

Assess networking requirements

Before providing S3 storage to clients, you must verify that networking is correctly configured to meet the S3 provisioning requirements.

Before you begin

The following cluster networking objects must be configured:

- Physical and logical ports
- Broadcast domains
- Subnets (if required)
- IPspaces (as required, in addition to the default IPspace)
- Failover groups (as required, in addition to the default failover group for each broadcast domain)
- External firewalls

About this task

For remote FabricPool capacity (cloud) tiers and remote S3 clients, you must use a data SVM and configure data LIFs. For FabricPool cloud tiers, you must also configure intercluster LIFs; cluster peering is not required.

For local FabricPool capacity tiers, you must use the system SVM (called “Cluster”), but you have two options for LIF configuration:

- You can use the cluster LIFs.

In this option, no further LIF configuration is required, but there will be an increase in traffic on the cluster LIFs. Also, the local tier will not be accessible to other clusters.

- You can use data and intercluster LIFs.

This option requires additional configuration, including enabling the LIFs for the S3 protocol, but the local tier will also be accessible as a remote FabricPool cloud tier to other clusters.

Steps

1. Display the available physical and virtual ports:

```
network port show
```

- When possible, you should use the port with the highest speed for the data network.
- All components in the data network must have the same MTU setting for best performance.

2. If you are planning to use a subnet name to allocate the IP address and network mask value for a LIF, verify that the subnet exists and has sufficient addresses available:

```
network subnet show
```

Subnets contain a pool of IP addresses that belong to the same layer 3 subnet. Subnets are created by using the `network subnet create` command.

3. Display available IPspaces:

```
network ipspace show
```

You can use the default IPspace or a custom IPspace.

4. If you want to use IPv6 addresses, verify that IPv6 is enabled on the cluster:

```
network options ipv6 show
```

If required, you can enable IPv6 by using the `network options ipv6 modify` command.

Decide where to provision new S3 storage capacity

Before you create a new S3 bucket, you must decide whether to place it in a new or existing SVM. This decision determines your workflow.

Choices

- If you want to provision a bucket in a new SVM or an SVM that is not enabled for S3, complete the steps in the following topics.

[Create an SVM for S3](#)

[Create a bucket for S3](#)

Although S3 can coexist in an SVM with NFS and SMB, you might choose to create a new SVM if one of the following is true:

- You are enabling S3 on a cluster for the first time.
- You have existing SVMs in a cluster in which you do not want to enable S3 support.
- You have one or more S3-enabled-SVMs in a cluster, and you want another S3 server with different performance characteristics.

After enabling S3 on the SVM, proceed to provision a bucket.

- If you want to provision the initial bucket or an additional bucket on an existing S3-enabled SVM, complete the steps in the following topic.

[Create a bucket for S3](#)

Configure S3 access to an SVM

Create an SVM for S3

Although S3 can coexist with other protocols in an SVM, you might want to create a new SVM to isolate the namespace and workload.

About this task

If you are only providing S3 object storage from an SVM, the S3 server does not require any DNS configuration. However, you might want to configure DNS on the SVM if other protocols are used.

When you configure S3 access to a new storage VM using System Manager, you are prompted to enter certificate and networking information, and the storage VM and S3 object storage server are created in a single operation.

Example 1. Steps

System Manager

You should be prepared to enter the S3 server name as a Fully Qualified Domain Name (FQDN), which clients will use for S3 access. The S3 server FQDN must not begin with a bucket name.


You should be prepared to enter IP addresses for interface role Data.

If you are using an external-CA signed certificate, you will be prompted to enter it during this procedure; you also have the option to use a system-generated certificate.

1. Enable S3 on a storage VM.

- a. Add a new storage VM: Click **Storage > Storage VMs**, then click **Add**.

If this is a new system with no existing storage VMs: Click **Dashboard > Configure Protocols**.

If you are adding an S3 server to an existing storage VM: Click **Storage > Storage VMs**, select a storage VM, click **Settings**, and then click  under **S3**.

- b. Click **Enable S3**, then enter the S3 Server Name.

- c. Select the certificate type.

Whether you select system-generated certificate or one of your own, it will be required for client access.

- d. Enter the network interfaces.

2. If you selected the system-generated certificate, you see the certificate information when the new storage VM creation is confirmed. Click **Download** and save it for client access.

- The secret key will not be displayed again.
- If you need the certificate information again: Click **Storage > Storage VMs**, select the storage VM, and click **Settings**.

CLI

1. Verify that S3 is licensed on your cluster:

```
system license show -package s3
```

If it is not, contact your sales representative.

2. Create an SVM:

```
vserver create -vserver <svm_name> -subtype default -rootvolume  
<root_volume_name> -aggregate <aggregate_name> -rootvolume-security  
-style unix -language C.UTF-8 -data-services <data-s3-server>  
-ipSPACE <ipSPACE_name>
```

- Use the UNIX setting for the `-rootvolume-security-style` option.

- Use the default C.UTF-8 -language option.
- The ipspace setting is optional.

3. Verify the configuration and status of the newly created SVM:

```
vserver show -vserver <svm_name>
```

The Vserver Operational State field must display the `running` state. If it displays the `initializing` state, it means that some intermediate operation such as root volume creation failed, and you must delete the SVM and re-create it.

Examples

The following command creates an SVM for data access in the IPspace ipspaceA:

```
cluster-1::> vserver create -vserver svm1.example.com -rootvolume  
root_svm1 -aggregate aggr1 -rootvolume-security-style unix -language  
C.UTF-8 -data-services _data-s3-server_ -ipspace ipspaceA
```

```
[Job 2059] Job succeeded:  
Vserver creation completed
```

The following command shows that an SVM was created with a root volume of 1 GB, and it was started automatically and is in `running` state. The root volume has a default export policy that does not include any rules, so the root volume is not exported upon creation. By default, the vsadmin user account is created and is in the `locked` state. The vsadmin role is assigned to the default vsadmin user account.


```

cluster-1::> vserver show -vserver svm1.example.com
                                Vserver: svm1.example.com
                                Vserver Type: data
                                Vserver Subtype: default
                                Vserver UUID: b8375669-19b0-11e5-b9d1-
00a0983d9736
                                Root Volume: root_svm1
                                Aggregate: aggr1
                                NIS Domain: -
                                Root Volume Security Style: unix
                                LDAP Client: -
                                Default Volume Language Code: C.UTF-8
                                Snapshot Policy: default
                                Comment:
                                Quota Policy: default
                                List of Aggregates Assigned: -
                                Limit on Maximum Number of Volumes allowed: unlimited
                                Vserver Admin State: running
                                Vserver Operational State: running
                                Vserver Operational State Stopped Reason: -
                                Allowed Protocols: nfs, cifs
                                Disallowed Protocols: -
                                QoS Policy Group: -
                                Config Lock: false
                                IPspace Name: ipspaceA

```

Create and install a CA certificate on the SVM

A Certificate Authority (CA) certificate is required to enable HTTPS traffic from S3 clients to the S3-enabled SVM.

About this task

Although it is possible to configure an S3 server to use HTTP only, and although it is possible to configure clients without a CA certificate requirement, it is a best practice to secure HTTPS traffic to ONTAP S3 servers with a CA certificate.

A CA certificate is not necessary for a local tiering use case, where IP traffic is going over cluster LIFs only.

The instructions in this procedure will create and install an ONTAP self-signed certificate. CA certificates from third-party vendors are also supported; see the administrator authentication documentation for more information.

Administrator authentication and RBAC

See the `security certificate` man pages for additional configuration options.

Steps

1. Create a self-signed digital certificate:

```
security certificate create -vserver svm_name -type root-ca -common-name  
ca_cert_name
```

The `-type root-ca` option creates and installs a self-signed digital certificate to sign other certificates by acting as a certificate authority (CA).

The `-common-name` option creates the SVM's Certificate Authority (CA) name and will be used when generating the certificate's complete name.

The default certificate size is 2048 bits.

Example

```
cluster-1::> security certificate create -vserver svm1.example.com -type  
root-ca -common-name svm1_ca
```

The certificate's generated name for reference:
svm1_ca_159D1587CE21E9D4_svm1_ca

When the certificate's generated name is displayed; be sure to save it for later steps in this procedure.

2. Generate a certificate signing request:

```
security certificate generate-csr -common-name s3_server_name  
[additional_options]
```

The `-common-name` parameter for the signing request must be the S3 server name (FQDN).

You can provide the location and other detailed information about the SVM if desired.

You are prompted to keep a copy of your certificate request and private key for future reference.

3. Sign the CSR using SVM_CA to generate S3 Server's certificate:

```
security certificate sign -vserver svm_name -ca ca_cert_name -ca-serial  
ca_cert_serial_number [additional_options]
```

Enter the command options that you used in previous steps:

- `-ca` — the common name of the CA that you entered in Step 1.
- `-ca-serial` — the CA serial number from Step 1. For example, if the CA certificate name is svm1_ca_159D1587CE21E9D4_svm1_ca, the serial number is 159D1587CE21E9D4.

By default, the signed certificate will expire in 365 days. You can select another value, and specify other signing details.

When prompted, copy and enter the certificate request string you saved in Step 2.

A signed certificate is displayed; save it for later use.

4. Install the signed certificate on the S3-enabled SVM:

```
security certificate install -type server -vserver svm_name
```

When prompted, enter the certificate and private key.

You have the option to enter intermediate certificates if a certificate chain is desired.

When the private key and the CA-signed digital certificate are displayed; save them for future reference.

5. Get the public key certificate:

```
security certificate show -vserver svm_name -common-name ca_cert_name -type  
root-ca -instance
```

Save the public key certificate for later client-side configuration.

Example

```
cluster-1::> security certificate show -vserver svm1.example.com -common  
-name svm1_ca -type root-ca -instance  
  
Name of Vserver: svm1.example.com  
FQDN or Custom Common Name: svm1_ca  
Serial Number of Certificate: 159D1587CE21E9D4  
Certificate Authority: svm1_ca  
Type of Certificate: root-ca  
(DEPRECATED)-Certificate Subtype: -  
Unique Certificate Name: svm1_ca_159D1587CE21E9D4_svm1_ca  
Size of Requested Certificate in Bits: 2048  
Certificate Start Date: Thu May 09 10:58:39 2020  
Certificate Expiration Date: Fri May 08 10:58:39 2021  
Public Key Certificate: -----BEGIN CERTIFICATE-----  
MIIDZ ...==  
-----END CERTIFICATE-----  
  
Country Name: US  
State or Province Name:  
Locality Name:  
Organization Name:  
Organization Unit:  
Contact Administrator's Email Address:  
Protocol: SSL  
Hashing Function: SHA256  
Self-Signed Certificate: true  
Is System Internal Certificate: false
```

Create an S3 service data policy

You can create service policies for S3 data and management services. An S3 service data policy is required to enable S3 data traffic on LIFs.

About this task

An S3 service data policy is required if you are using data LIFs and intercluster LIFs. It is not required if you are using cluster LIFs for the local tiering use case.

When a service policy is specified for a LIF, the policy is used to construct a default role, failover policy, and data protocol list for the LIF.

Although multiple protocols can be configured for SVMs and LIFs, it is a best practice for S3 to be the only protocol when serving object data.

Steps

1. Change the privilege setting to advanced:

```
set -privilege advanced
```

2. Create a service data policy:

```
network interface service-policy create -vserver svm_name -policy policy_name  
-services data-core,data-s3-server
```

The `data-core` and `data-s3-server` services are the only ones required to enable ONTAP S3, although other services can be included as needed.

Create data LIFs

If you created a new SVM, the dedicated LIFs you create for S3 access should be data LIFs.

Before you begin

- The underlying physical or logical network port must have been configured to the administrative `up` status.
- If you are planning to use a subnet name to allocate the IP address and network mask value for a LIF, the subnet must already exist.

Subnets contain a pool of IP addresses that belong to the same layer 3 subnet. They are created using the `network subnet create` command.

- The LIF service policy must already exist.

About this task

- You can create both IPv4 and IPv6 LIFs on the same network port.
- If you have a large number of LIFs in your cluster, you can verify the LIF capacity supported on the cluster by using the `network interface capacity show` command and the LIF capacity supported on each node by using the `network interface capacity details show` command (at the advanced privilege level).
- If you are enabling remote FabricPool capacity (cloud) tiering, you must also configure intercluster LIFs.

Steps

1. Create a LIF:

```
network interface create -vserver svm_name -lif lif_name -service-policy
service_policy_names -home-node node_name -home-port port_name {-address
IP_address -netmask IP_address | -subnet-name subnet_name} -firewall-policy
data -auto-revert {true|false}
```

- `-home-node` is the node to which the LIF returns when the `network interface revert` command is run on the LIF.

You can also specify whether the LIF should automatically revert to the home-node and home-port with the `-auto-revert` option.

- `-home-port` is the physical or logical port to which the LIF returns when the `network interface revert` command is run on the LIF.
- You can specify an IP address with the `-address` and `-netmask` options, or you enable allocation from a subnet with the `-subnet_name` option.
- When using a subnet to supply the IP address and network mask, if the subnet was defined with a gateway, a default route to that gateway is added automatically to the SVM when a LIF is created using that subnet.
- If you assign IP addresses manually (without using a subnet), you might need to configure a default route to a gateway if there are clients or domain controllers on a different IP subnet. The `network route create` man page contains information about creating a static route within an SVM.
- For the `-firewall-policy` option, use the same default data as the LIF role.

You can create and add a custom firewall policy later if desired.



Beginning with ONTAP 9.10.1, firewall policies are deprecated and wholly replaced with LIF service policies. For more information, see [Configure firewall policies for LIFs](#).

- `-auto-revert` allows you to specify whether a data LIF is automatically reverted to its home node under circumstances such as startup, changes to the status of the management database, or when the network connection is made. The default setting is `false`, but you can set it to `true` depending on network management policies in your environment.
- The `-service-policy` option specifies the data and management services policy you created and any other policies you need.

2. If you want to assign an IPv6 address in the `-address` option:

- a. Use the `network ndp prefix show` command to view the list of RA prefixes learned on various interfaces.

The `network ndp prefix show` command is available at the advanced privilege level.

- b. Use the format `prefix:id` to construct the IPv6 address manually.

`prefix` is the prefix learned on various interfaces.

For deriving the `id`, choose a random 64-bit hexadecimal number.

3. Verify that the LIF was created successfully by using the `network interface show` command.
4. Verify that the configured IP address is reachable:

To verify an...	Use...
IPv4 address	<code>network ping</code>
IPv6 address	<code>network ping6</code>

Examples

The following command shows how to create an S3 data LIF that is assigned with the `my-S3-policy` service policy:

```
network interface create -vserver svml.example.com -lif lif2 -home-node  
node2 -homeport e0d -service-policy my-S3-policy -subnet-name ipspace1
```

The following command shows all the LIFs in cluster-1. Data LIFs `datalif1` and `datalif3` are configured with IPv4 addresses, and `datalif4` is configured with an IPv6 address:

```
cluster-1::> network interface show
```

Vserver	Logical Interface	Status Admin/Oper	Network Address/Mask	Current Node	Current Is Port
Home					
-----	-----	-----	-----	-----	-----
cluster-1	cluster_mgmt	up/up	192.0.2.3/24	node-1	e1a
true					
node-1	clus1	up/up	192.0.2.12/24	node-1	e0a
true					
	clus2	up/up	192.0.2.13/24	node-1	e0b
true					
	mgmt1	up/up	192.0.2.68/24	node-1	e1a
true					
node-2	clus1	up/up	192.0.2.14/24	node-2	e0a
true					
	clus2	up/up	192.0.2.15/24	node-2	e0b
true					
	mgmt1	up/up	192.0.2.69/24	node-2	e1a
true					
vs1.example.com	datalif1	up/down	192.0.2.145/30	node-1	e1c
true					
vs3.example.com	datalif3	up/up	192.0.2.146/30	node-2	e0c
true					
	datalif4	up/up	2001::2/64	node-2	e0c
true					

5 entries were displayed.

Create intercluster LIFs for remote FabricPool tiering

If you are enabling remote FabricPool capacity (cloud) tiering using ONTAP S3, you must configure intercluster LIFs. You can configure intercluster LIFs on ports shared with the data network. Doing so reduces the number of ports you need for intercluster networking.

Before you begin

- The underlying physical or logical network port must have been configured to the administrative up status.
- The LIF service policy must already exist.

About this task

Intercluster LIFs are not required for local Fabric pool tiering or for serving external S3 apps.

Steps

1. List the ports in the cluster:

```
network port show
```

The following example shows the network ports in `cluster01`:

```
cluster01::> network port show
```

						Speed
(Mbps)						
Node	Port	IPspace	Broadcast Domain	Link	MTU	Admin/Oper
-----	-----	-----	-----	-----	-----	
cluster01-01						
	e0a	Cluster	Cluster	up	1500	auto/1000
	e0b	Cluster	Cluster	up	1500	auto/1000
	e0c	Default	Default	up	1500	auto/1000
	e0d	Default	Default	up	1500	auto/1000
cluster01-02						
	e0a	Cluster	Cluster	up	1500	auto/1000
	e0b	Cluster	Cluster	up	1500	auto/1000
	e0c	Default	Default	up	1500	auto/1000
	e0d	Default	Default	up	1500	auto/1000

2. Create intercluster LIFs on the system SVM:

```
network interface create -vserver Cluster -lif LIF_name -service-policy  
default-intercluster -home-node node -home-port port -address port_IP -netmask  
netmask
```

The following example creates intercluster LIFs `cluster01_icl01` and `cluster01_icl02`:

```
cluster01::> network interface create -vserver Cluster -lif  
cluster01_icl01 -service-  
policy default-intercluster -home-node cluster01-01 -home-port e0c  
-address 192.168.1.201  
-netmask 255.255.255.0  
  
cluster01::> network interface create -vserver Cluster -lif  
cluster01_icl02 -service-  
policy default-intercluster -home-node cluster01-02 -home-port e0c  
-address 192.168.1.202  
-netmask 255.255.255.0
```


3. Verify that the intercluster LIFs were created:

```
network interface show -service-policy default-intercluster
```

```
cluster01::> network interface show -service-policy default-intercluster
```

	Logical	Status	Network	Current	
Current Is					
Vserver	Interface	Admin/Oper	Address/Mask	Node	Port
Home					
-----	-----	-----	-----	-----	-----
cluster01	cluster01_icl01	up/up	192.168.1.201/24	cluster01-01	e0c
true	cluster01_icl02	up/up	192.168.1.202/24	cluster01-02	e0c
true					

4. Verify that the intercluster LIFs are redundant:

```
network interface show -service-policy default-intercluster -failover
```

The following example shows that the intercluster LIFs cluster01_icl01 and cluster01_icl02 on the e0c port will fail over to the e0d port.

```
cluster01::> network interface show -service-policy default-intercluster -failover
```

	Logical	Home	Failover	Failover
Vserver	Interface	Node:Port	Policy	Group
-----	-----	-----	-----	-----
cluster01	cluster01_icl01	cluster01-01:e0c	local-only	
192.168.1.201/24				
		Failover Targets: cluster01-01:e0c,		
		cluster01-01:e0d		
	cluster01_icl02	cluster01-02:e0c	local-only	
192.168.1.201/24				
		Failover Targets: cluster01-02:e0c,		
		cluster01-02:e0d		

Create the S3 object store server

The ONTAP object store server manages data as S3 objects, as opposed to file or block storage provided by ONTAP NAS and SAN servers.

Before you begin

You should be prepared to enter the S3 server name as a Fully Qualified Domain Name (FQDN), which clients will use for S3 access. The FQDN must not begin with a bucket name.

You should have a self-signed CA certificate (created in previous steps) or a certificate signed by an external CA vendor. A CA certificate is not necessary for a local tiering use case, where IP traffic is going over cluster LIFs only.

About this task

When an object store server is created, a root user with UID 0 is created. No access key or secret key is generated for this root user. The ONTAP administrator must run the `object-store-server users regenerate-keys` command to set the access key and secret key for this user.



As a NetApp best practice, do not use this root user. Any client application that uses the access key or secret key of the root user has full access to all buckets and objects in the object store.

See the `vserver object-store-server` man pages for additional configuration and display options.


Example 2. Steps

System Manager

Use this procedure if you are adding an S3 server to an existing storage VM. To add an S3 server to a new storage VM, see [Create a storage SVM for S3](#).

You should be prepared to enter IP addresses for interface role Data.

1. Enable S3 on an existing storage VM.

- a. Select the storage VM: click **Storage > Storage VMs**, select a storage VM, click **Settings**, and then click  under **S3**.
- b. Click **Enable S3**, then enter the S3 Server Name.
- c. Select the certificate type.

Whether you select system-generated certificate or one of your own, it will be required for client access.

- d. Enter the network interfaces.

2. If you selected the system-generated certificate, you see the certificate information when the new storage VM creation is confirmed. Click **Download** and save it for client access.

- The secret key will not be displayed again.
- If you need the certificate information again: click **Storage > Storage VMs**, select the storage VM, and click **Settings**.

CLI

1. Create the S3 server:

```
vserver object-store-server create -vserver svm_name -object-store-server  
s3_server_fqdn -certificate-name server_certificate_name -comment text  
[additional_options]
```

You can specify additional options when creating the S3 server or at any time later.

- If you are configuring local tiering, the SVM name can either be a data SVM or system SVM (cluster) name.
- The certificate name should be the name of the server certificate (end user or leaf certificate), and not server CA certificate (intermediate or root CA certificate).
- HTTPS is enabled by default on port 443. You can change the port number with the `-secure -listener-port` option.

When HTTPS is enabled, CA certificates are required for correct integration with SSL/TLS.

- HTTP is disabled by default. When enabled, the server listens on port 80. You can enable it with the `-is-http-enabled` option, or change the port number with the `-listener-port` option.

When HTTP is enabled, the request and responses are sent over the network in clear text.

2. Verify that S3 is configured:

```
vserver object-store-server show
```

Example

This command verifies the configuration values of all object storage servers:

```
cluster1::> vserver object-store-server show

Vserver: vs1

Object Store Server Name: s3.example.com
Administrative State: up
Listener Port For HTTP: 80
Secure Listener Port For HTTPS: 443
HTTP Enabled: false
HTTPS Enabled: true
Certificate for HTTPS Connections: svm1_ca
Comment: Server comment
```

Add storage capacity to an S3-enabled SVM

Create a bucket

S3 objects are kept in *buckets*. They are not nested as files inside a directory inside other directories.

Before you begin

A storage VM containing an S3 server must already exist.

About this task

- Beginning with ONTAP 9.14.1, automatic resizing has been enabled on S3 FlexGroup volumes when buckets are created on them. This eliminates excessive capacity allocation during bucket creation on existing and new FlexGroup volumes. FlexGroup volumes are resized to a minimum required size based on the following guidelines. The minimum required size is the total size of all the S3 buckets in a FlexGroup volume.
 - Beginning with ONTAP 9.14.1, if an S3 FlexGroup volume is created as part of a new bucket creation, the FlexGroup volume is created with the minimum required size.
 - If an S3 FlexGroup volume was created prior to ONTAP 9.14.1, the first bucket created or deleted subsequent to ONTAP 9.14.1 resizes the FlexGroup volume to the minimum required size.
 - If an S3 FlexGroup volume was created prior to ONTAP 9.14.1, and already had the minimum required size, the creation or deletion of a bucket subsequent to ONTAP 9.14.1 maintains the size of the S3 FlexGroup volume.
- Storage service levels are predefined adaptive Quality of Service (QoS) policy groups, with *value*, *performance*, and *extreme* default levels. Instead of one of the default storage service levels, you can also define a custom QoS policy group and apply it to a bucket. For more information about storage service definitions, see [Storage service definitions](#). For more information about performance management, see [Performance management](#).
Beginning with ONTAP 9.8, when you provision storage, QoS is enabled by default. You can disable QoS or choose a custom QoS policy during the provisioning process or at a later time.

- If you are configuring local capacity tiering, you create buckets and users in a data storage VM, not in the system storage VM where the S3 server is located.
 - For remote client access, you must configure buckets in an S3-enabled storage VM. If you create a bucket in a storage VM that is not S3-enabled, it will only be available for local tiering.
 - Beginning with ONTAP 9.14.1, you can [create a bucket on a mirrored or unmirrored aggregate in a MetroCluster configuration](#).
 - For the CLI, when you create a bucket, you have two provisioning options:
 - Let ONTAP select the underlying aggregates and FlexGroup components (default)
 - ONTAP creates and configures a FlexGroup volume for the first bucket by automatically selecting the aggregates. It will automatically select the highest service level available for your platform, or you can specify the storage service level. Any additional buckets you add later in the storage VM will have the same underlying FlexGroup volume.
 - Alternatively, you can specify whether the bucket will be used for tiering, in which case ONTAP tries to select low-cost media with optimal performance for the tiered data.
 - You select the underlying aggregates and FlexGroup components (requires advanced privilege command options): You have the option to manually select the aggregates on which the bucket and containing FlexGroup volume must be created, and then specifying the number of constituents on each aggregate. When adding additional buckets:
 - If you specify aggregates and constituents for a new bucket, a new FlexGroup will be created for the new bucket.
 - If you do not specify aggregates and constituents for a new bucket, the new bucket will be added to an existing FlexGroup.
- See [FlexGroup volumes management](#) for more information.

When you specify aggregates and constituents when creating a bucket, no QoS policy groups, default or custom, are applied. You can do so later with the `vserver object-store-server bucket modify` command.

See [vserver object-store-server bucket modify](#) for more information.

Note: If you are serving buckets from Cloud Volumes ONTAP, you should use the CLI procedure. It is strongly recommended that you manually select the underlying aggregates to ensure that they are using one node only. Using aggregates from both nodes can impact performance, because the nodes will be in geographically separated availability zones and hence susceptible to latency issues.

Create S3 buckets with the ONTAP CLI

1. If you plan to select aggregates and FlexGroup components yourself, set the privilege level to advanced (otherwise, admin privilege level is sufficient): `set -privilege advanced`
2. Create a bucket:

```
vserver object-store-server bucket create -vserver svm_name -bucket
bucket_name [-size integer[KB|MB|GB|TB|PB]] [-comment text]
[additional_options]
```

The storage VM name can be either a data storage VM or `Cluster` (the system storage VM name) if you are configuring local tiering.

If you specify no options, ONTAP creates an 800GB bucket with the service level set to the highest level available for your system.

If you want ONTAP to create a bucket based on performance or usage, use one of the following options:

- service level

Include the `-storage-service-level` option with one of the following values: `value`, `performance`, or `extreme`.

- tiering

Include the `-used-as-capacity-tier true` option.

If you want to specify the aggregates on which to create the underlying FlexGroup volume, use the following options:

- The `-aggr-list` parameter specifies the list of aggregates to be used for FlexGroup volume constituents.

Each entry in the list creates a constituent on the specified aggregate. You can specify an aggregate multiple times to have multiple constituents created on the aggregate.

For consistent performance across the FlexGroup volume, all of the aggregates must use the same disk type and RAID group configurations.

- The `-aggr-list-multiplier` parameter specifies the number of times to iterate over the aggregates that are listed with the `-aggr-list` parameter when creating a FlexGroup volume.

The default value of the `-aggr-list-multiplier` parameter is 4.

3. Add a QoS policy group if needed:

```
vserver object-store-server bucket modify -bucket bucket_name -qos-policy  
-group qos_policy_group
```

4. Verify bucket creation:

```
vserver object-store-server bucket show [-instance]
```

Example

The following example creates a bucket for storage VM `vs1` of size 1TB and specifying the aggregate:

```
cluster-1::*> vserver object-store-server bucket create -vserver  
svml.example.com -bucket testbucket -aggr-list aggr1 -size 1TB
```

Create S3 buckets with System Manager

1. Add a new bucket on an S3-enabled storage VM.
 - a. Click **Storage > Buckets**, then click **Add**.
 - b. Enter a name, select the storage VM, and enter a size.

- If you click **Save** at this point, a bucket is created with these default settings:
 - No users are granted access to the bucket unless any group policies are already in effect.



You should not use the S3 root user to manage ONTAP object storage and share its permissions, because it has unlimited access to the object store. Instead, create a user or group with administrative privileges that you assign.

- A Quality of Service (performance) level that is the highest available for your system.
- Click **Save** to create a bucket with these default values.

Configure additional permissions and restrictions

You can click **More Options** to configure settings for object locking, user permissions, and performance level when you configure the bucket, or you can modify these settings later.

If you intend to use the S3 object store for FabricPool tiering, consider selecting **Use for tiering** (use low-cost media with optimal performance for the tiered data) rather than a performance service level.

If you want to enable versioning for your objects for later recovery, select **Enable Versioning**. Versioning is enabled by default if you are enabling object locking on the bucket. For information about object versioning, see the [Using versioning in S3 buckets for Amazon](#).

Beginning with 9.14.1, object locking is supported on S3 buckets. S3 object locking requires a standard SnapLock license. This license is included with [ONTAP One](#).

Prior to ONTAP One, the SnapLock license was included in the Security and Compliance bundle. The Security and Compliance bundle is no longer offered but is still valid. Although not currently required, existing customers can choose to [upgrade to ONTAP One](#).

If you are enabling object locking on a bucket, you should [verify that a SnapLock license is installed](#). If a SnapLock license is not installed, you must [install](#) it before you can enable object locking.

When you have verified that the SnapLock license is installed, to protect objects in your bucket from getting deleted or overwritten, select **Enable object locking**. Locking can be enabled on either all or specific versions of objects, and only when the SnapLock compliance clock is initialized for the cluster nodes. Follow these steps:

1. If the SnapLock compliance clock is not initialized on any node of the cluster, the **Initialize SnapLock Compliance Clock** button appears. Click **Initialize SnapLock Compliance Clock** to initialize the SnapLock compliance clock on the cluster nodes.
2. Select **Governance** mode to activate a time-based lock that allows *Write once, read many (WORM)* permissions on the objects. Even in *Governance* mode, the objects can be deleted by administrator users with specific permissions.
3. Select **Compliance** mode if you want to assign stricter rules of deletion and update on the objects. In this mode of object locking, the objects can be expired only on the completion of the specified retention period. Unless a retention period is specified, the objects remain locked indefinitely.
4. Specify the retention tenure for the lock in days or years if you want the locking to be effective for a certain period.



Locking is applicable to versioned and non-versioned S3 buckets. Object locking is not applicable to NAS objects.

You can configure protection and permission settings, and performance service level for the bucket.



You must have already created user and groups before configuring the permissions.

For information, see [Create mirror for new bucket](#).

Verify access to the bucket

On S3 client applications (whether ONTAP S3 or an external third-party application), you can verify your access to the newly created bucket by entering the following:

- The S3 server CA certificate.
- The user's access key and secret key.
- The S3 server FQDN name and bucket name.

Create a bucket on a mirrored or unmirrored aggregate in a MetroCluster configuration

Beginning with ONTAP 9.14.1, you can provision a bucket on a mirrored or unmirrored aggregate in MetroCluster FC and IP configurations.

About this task

- By default, buckets are provisioned on mirrored aggregates.
- The same provisioning guidelines outlined in [Create a bucket](#) apply to creating a bucket in a MetroCluster environment.
- The following S3 object storage features are **not** supported in MetroCluster environments:
 - S3 SnapMirror
 - S3 bucket lifecycle management
 - S3 object lock in **Compliance** mode



S3 object lock in **Governance** mode is supported.

- Local FabricPool tiering

Before you begin

An SVM containing an S3 server must already exist.

Process to create buckets

CLI

1. If you plan to select aggregates and FlexGroup components yourself, set the privilege level to advanced (otherwise, admin privilege level is sufficient): `set -privilege advanced`
2. Create a bucket:

```
vserver object-store-server bucket create -vserver <svm_name> -bucket  
<bucket_name> [-size integer[KB|MB|GB|TB|PB]] [-use-mirrored-aggregates  
true/false]
```

Set the `-use-mirrored-aggregates` option to `true` or `false` depending on whether you want to use a mirrored or unmirrored aggregate.



By default, the `-use-mirrored-aggregates` option is set to `true`.

- The SVM name must be a data SVM.
- If you specify no options, ONTAP creates an 800GB bucket with the service level set to the highest level available for your system.
- If you want ONTAP to create a bucket based on performance or usage, use one of the following options:
 - **service level**

Include the `-storage-service-level` option with one of the following values: `value`, `performance`, or `extreme`.
 - **tiering**

Include the `-used-as-capacity-tier true` option.
- If you want to specify the aggregates on which to create the underlying FlexGroup volume, use the following options:
 - The `-aggr-list` parameter specifies the list of aggregates to be used for FlexGroup volume constituents.

Each entry in the list creates a constituent on the specified aggregate. You can specify an aggregate multiple times to have multiple constituents created on the aggregate.

For consistent performance across the FlexGroup volume, all of the aggregates must use the same disk type and RAID group configurations.
 - The `-aggr-list-multiplier` parameter specifies the number of times to iterate over the aggregates that are listed with the `-aggr-list` parameter when creating a FlexGroup volume.

The default value of the `-aggr-list-multiplier` parameter is 4.

3. Add a QoS policy group if needed:

```
vserver object-store-server bucket modify -bucket bucket_name -qos-policy  
-group qos_policy_group
```

4. Verify bucket creation:

```
vserver object-store-server bucket show [-instance]
```

Example

The following example creates a bucket for SVM vs1 of size 1TB on a mirrored aggregate:

```
cluster-1::*> vserver object-store-server bucket create -vserver  
svml.example.com -bucket testbucket -size 1TB -use-mirrored-aggregates  
true
```


System Manager

1. Add a new bucket on an S3-enabled storage VM.
 - a. Click **Storage > Buckets**, then click **Add**.
 - b. Enter a name, select the storage VM, and enter a size.

By default, the bucket is provisioned on a mirrored aggregate. If you want to create a bucket on an unmirrored aggregate, select **More Options** and uncheck the **Use the SyncMirror tier** box under **Protection** as shown in the following image:

Add bucket ✕

NAME

 To use this bucket from a remote cluster, configure S3 service on storage VM "vs1".

FOLDER (OPTIONAL)

Specify the folder to map to this bucket. [Know more](#)

CAPACITY

Size

GB

☐ Use tiering
If you select this option, the system will try to select low-cost media with optimal performance for the tiered data.
☐ Enable versioning
Versioning-enabled buckets allow you to recover objects that were accidentally deleted or overwritten. After versioning is enabled, it can't be disabled. However, you can suspend versioning.

PERFORMANCE SERVICE LEVEL

Value

Not sure? [Get help selecting type](#)

Permissions

☐ Copy access permissions from an existing bucket

Principal	Effect	Actions	Resources	Conditions
All users of this stor...	allow	ListBucket	*	

+ Add

Object locking

☐ Enable object locking

Object locking utilizes the "Write Once, Read Many" (WORM) model in which objects or their versions are protected from being deleted or overwritten during the specified retention period.

Protection

☒ Use the S3x3lination

- If you click **Save** at this point, a bucket is created with these default settings:
 - No users are granted access to the bucket unless any group policies are already in effect.



You should not use the S3 root user to manage ONTAP object storage and share its permissions, because it has unlimited access to the object store. Instead, create a user or group with administrative privileges that you assign.

- A Quality of Service (performance) level that is the highest available for your system.
- You can click **More Options** to configure user permissions and performance level when you configure the bucket, or you can modify these settings later.
 - You must have already created user and groups before using **More Options** to configure their permissions.

- If you intend to use the S3 object store for FabricPool tiering, consider selecting **Use for tiering** (use low-cost media with optimal performance for the tiered data) rather than a performance service level.
2. On S3 client apps – another ONTAP system or an external 3rd-party app – verify access to the new bucket by entering the following:
 - The S3 server CA certificate.
 - The user's access key and secret key.
 - The S3 server FQDN name and bucket name.

Create a bucket lifecycle management rule

Beginning with ONTAP 9.13.1, you can create lifecycle management rules to manage object lifecycles in your S3 buckets. You can define deletion rules for specific objects in a bucket, and through these rules, expire those bucket objects. This enables you to meet retention requirements and manage overall S3 object storage efficiently.



If object locking is enabled for your bucket objects, the lifecycle management rules for object expiration will not be applied on locked objects. For information about object locking, see [Create a bucket](#).

Before you begin

An S3-enabled SVM containing an S3 server and a bucket must already exist. See [Create an SVM for S3](#) for more information.

About this task

When creating your lifecycle management rules, you can apply the following deletion actions to your bucket objects:

- Deletion of current versions - This action expires objects identified by the rule. If versioning is enabled on the bucket, S3 makes all expired objects unavailable. If versioning is not enabled, this rule deletes the objects permanently. The CLI action is `Expiration`.
- Deletion of non-current versions - This action specifies when S3 can permanently remove non-current objects. The CLI action is `NoncurrentVersionExpiration`.
- Deletion of expired delete markers - This action deletes expired object delete markers. In versioning-enabled buckets, objects with a delete markers become the current versions of the objects. The objects are not deleted, and no action can be performed on them. These objects become expired when there are no current versions associated with them. The CLI action is `Expiration`.
- Deletion of incomplete multipart uploads - This action sets a maximum time (in days) that you want to allow multipart uploads to remain in progress. Following which, they are deleted. The CLI action is `AbortIncompleteMultipartUpload`.

The procedure you follow depends on the interface that you use. With ONTAP 9.13.1, you need to use the CLI. Beginning with ONTAP 9.14.1, you can also use System Manager.

Manage lifecycle management rules with the CLI

Beginning with ONTAP 9.13.1, you can use the ONTAP CLI to create lifecycle management rules to expire objects in your S3 buckets.

Before you begin

For the CLI, you need to define the required fields for each expiration action type when creating a bucket lifecycle management rule. These fields can be modified after initial creation. The following table displays the unique fields for each action type.

Action type	Unique fields
NonCurrentVersionExpiration	<ul style="list-style-type: none">• <code>-non-curr-days</code> - Number of days after which non-current versions will be deleted• <code>-new-non-curr-versions</code> - Number of latest non-current versions to be retained
Expiration	<ul style="list-style-type: none">• <code>-obj-age-days</code> - Number of days since creation, after which current version of objects can be deleted• <code>-obj-exp-date</code> - Specific date when the objects should expire• <code>-expired-obj-del-markers</code> - Cleanup object delete markers
AbortIncompleteMultipartUpload	<ul style="list-style-type: none">• <code>-after-initiation-days</code> - Number of days of initiation, after which upload can be aborted

In order for the bucket lifecycle management rule to only be applied to a specific subset of objects, admins must set each filter when creating the rule. If these filters are not set when creating the rule, the rule will be applied to all objects within the bucket.

All filters can be modified after initial creation *except* for the following: +

- `-prefix`
- `-tags`
- `-obj-size-greater-than`
- `-obj-size-less-than`

Steps

1. Use the `vserver object-store-server bucket lifecycle-management-rule create` command with required fields for your expiration action type to create your bucket lifecycle management rule.

Example

The following command creates a `NonCurrentVersionExpiration` bucket lifecycle management rule:

```
vserver object-store-server bucket lifecycle-management-rule create
-vserver <svm_name> -bucket <bucket_name> -rule-id <rule_name> -action
NonCurrentVersionExpiration -index <lifecycle_rule_index_integer> -is
-enabled {true|false} -prefix <object_name> -tags <text> -obj-size-greater
-than {<integer>[KB|MB|GB|TB|PB]} -obj-size-less-than
{<integer>[KB|MB|GB|TB|PB]} -new-non-curr-versions <integer> -non-curr
-days <integer>
```

Example

The following command creates an Expiration bucket lifecycle management rule:

```
vserver object-store-server bucket lifecycle-management-rule create
-vserver <svm_name> -bucket <bucket_name> -rule-id <rule_name> -action
Expiration -index <lifecycle_rule_index_integer> -is-enabled {true|false}
-prefix <object_name> -tags <text> -obj-size-greater-than
{<integer>[KB|MB|GB|TB|PB]} -obj-size-less-than
{<integer>[KB|MB|GB|TB|PB]} -obj-age-days <integer> -obj-exp-date
<"MM/DD/YYYY HH:MM:SS"> -expired-obj-del-marker {true|false}
```

Example


The following command creates an AbortIncompleteMultipartUpload bucket lifecycle management rule:

```
vserver object-store-server bucket lifecycle-management-rule create
-vserver <svm_name> -bucket <bucket_name> -rule-id <rule_name> -action
AbortIncompleteMultipartUpload -index <lifecycle_rule_index_integer> -is
-enabled {true|false} -prefix <object_name> -tags <text> -obj-size-greater
-than {<integer>[KB|MB|GB|TB|PB]} -obj-size-less-than
{<integer>[KB|MB|GB|TB|PB]} -after-initiation-days <integer>
```

Manage lifecycle management rules with System Manager

Beginning with ONTAP 9.14.1, you can expire S3 objects by using System Manager. You can add, edit, and delete lifecycle management rules for your S3 objects. Additionally, you can import a lifecycle rule created for one bucket and utilize it for the objects in another bucket. You can disable an active rule and enable it later.

Add a lifecycle management rule

1. Click **Storage > Buckets**.
2. Select the bucket for which you want to specify the expiration rule.
3. Click the  icon and select **Manage lifecycle rules**.
4. Click **Add > Lifecycle rule**.
5. On the Add a lifecycle rule page, add the name of the rule.
6. Define the scope of the rule, whether you want it to apply to all the objects in the bucket or on specific

objects. If you want to specify objects, add at least one of the following filter criteria:

- a. **Prefix:** Specify a prefix of the object key names to which the rule should apply. Typically, it is the path or folder of the object. You can enter one prefix per rule. Unless a valid prefix is provided, the rule applies to all the objects in a bucket.
- b. **Tags:** Specify up to three key and value pairs (tags) for the objects to which the rule should apply. Only valid keys are used for filtering. The value is optional. However, if you add values, ensure that you add only valid values for the corresponding keys.
- c. **Size:** You can limit the scope between the minimum and maximum sizes of the objects. You can enter either or both the values. The default unit is MiB.

7. Specify the action:


- a. **Expire the current version of objects:** Set a rule to make all current objects permanently unavailable after a specific number of days since their creation, or on a specific date. This option is unavailable if the **Delete expired object delete markers** option is selected.
- b. **Permanently delete noncurrent versions:** Specify the number of days after which the version becomes non-current, and thereafter can be deleted, and the number of versions to retain.
- c. **Delete expired object delete markers:** Select this action to delete objects with expired delete markers, that is delete markers without an associated current object.



This option becomes unavailable when you select the **Expire the current version of objects** option that automatically deletes all objects after the retention period. This option also becomes unavailable when object tags are used for filtering.

- d. **Delete incomplete multipart uploads:** Set the number of days after which incomplete multipart uploads are to be deleted. If the multipart uploads that are in progress fail within the specified retention period, you can delete the incomplete multipart uploads. This option becomes unavailable when object tags are used for filtering.
- e. Click **Save**.


Import a lifecycle rule

1. Click **Storage > Buckets**.
2. Select the bucket for which you want to import the expiration rule.
3. Click the  icon and select **Manage lifecycle rules**.
4. Click **Add > Import a rule**.
5. Select the bucket from which you want to import the rule. The lifecycle management rules defined for the selected bucket appear.
6. Select the rule that you want to import. You have the option to select one rule at a time, with the default selection being the first rule.
7. Click **Import**.

Edit, delete, or disable a rule

You can only edit the lifecycle management actions associated with the rule. If the rule was filtered with object tags, then the **Delete expired object delete markers** and **Delete incomplete multipart uploads** options are unavailable.

When you delete a rule, that rule will no longer apply to previously associated objects.

1. Click **Storage > Buckets**.
2. Select the bucket for which you want to edit, delete, or disable the lifecycle management rule.
3. Click the  icon and select **Manage lifecycle rules**.
4. Select the required rule. You can edit and disable one rule at a time. You can delete multiple rules at once.
5. Select **Edit**, **Delete**, or **Disable**, and complete the procedure.

Create an S3 user

User authorization is required on all ONTAP object stores to restrict connectivity to authorized clients.

Before you begin.

An S3-enabled storage VM must already exist.

About this task

An S3 user can be granted access to any bucket in a storage VM. When you create an S3 user, an access key and a secret key are also generated for the user. They should be shared with the user along with the FQDN of the object store and bucket name. An S3 users' keys can be viewed with the `vserver object-store-server user show` command.

You can grant specific access permissions to S3 users in a bucket policy or an object server policy.



When you create a new object store server, ONTAP creates a root user (UID 0), which is a privileged user with access to all buckets. Rather than administering ONTAP S3 as the root user, NetApp recommends that an admin user role be created with specific privileges.

CLI

1. Create an S3 user:

```
vserver object-store-server user create -vserver svm_name -user user_name  
-comment [-comment text] -key-time-to-live time
```


- Adding a comment is optional.
- Beginning with ONTAP 9.14.1, you can define the period of time for which the key will be valid in the `-key-time-to-live` parameter. You can add the retention period in this format, to indicate the period after which the access key expires:
`P[<integer>D]T[<integer>H][<integer>M][<integer>S] | P<integer>W`
For example, if you want to enter a retention period of one day, two hours, three minutes, and four seconds, enter the value as `P1DT2H3M4S`. Unless specified, the key is valid for an indefinite period of time.

The below example creates a user with name `sm_user1` on storage VM `vs0`, with a key retention period of one week.

```
vserver object-store-server user create -vserver vs0 -user  
sm_user1 -key-time-to-live P1W
```

2. Be sure to save the access key and secret key. They will be required for access from S3 clients.

System Manager

1. Click **Storage > Storage VMs**. Select the storage VM to which you need to add a user, select **Settings** and then click  under S3.
2. To add a user, click **Users > Add**.
3. Enter a name for the user.
4. Beginning with ONTAP 9.14.1, you can specify the retention period of the access keys that get created for the user. You can specify the retention period in days, hours, minutes, or seconds, after which the keys automatically expire. By default, the value is set to 0 that indicates that the key is indefinitely valid.
5. Click **Save**. The user is created, and an access key and a secret key are generated for the user.
6. Download or save the access key and secret key. They will be required for access from S3 clients.

Next steps

- [Create or modify S3 groups](#)

Create or modify S3 groups

You can simplify bucket access by creating groups of users with appropriate access authorizations.

Before you begin

S3 users in an S3-enabled SVM must already exist.

About this task

Users in an S3 group can be granted access to any bucket in an SVM but not in multiple SVMs. Group access permissions can be configured in two ways:


- At the bucket level

After creating a group of S3 users, you specify group permissions in bucket policy statements and they apply only to that bucket.

- At the SVM level

After creating a group of S3 users, you specify object server policy names in the group definition. Those policies determine the buckets and access for the group members.

System Manager

1. Edit the storage VM: click **Storage > storage VMs**, click the storage VM, click **Settings** and then click  under S3.
2. Add a group: select **Groups**, then select **Add**.
3. Enter a group name and select from a list of users.
4. You can select an existing group policy or add one now, or you can add a policy later.

CLI

1. Create an S3 group:

```
vserver object-store-server group create -vserver svm_name -name group_name  
-users user_name\(s\) [-policies policy_names] [-comment text\]
```

The `-policies` option can be omitted in configurations with only one bucket in an object store; the group name can be added to the bucket policy.

The `-policies` option can be added later with the `vserver object-store-server group modify` command after object storage server policies are created.

Regenerate keys and modify their retention period

Access keys and secret keys are automatically generated during user creation for enabling S3 client access. You can regenerate keys for a user if a key is expired or compromised.

For information about generation of access keys, see [Create an S3 user](#).



CLI

1. Regenerate access and secret keys for a user by running the `vserver object-store-server user regenerate-keys` command.
2. By default, generated keys are valid indefinitely. Beginning with 9.14.1, you can modify their retention period, after which the keys automatically expire. You can add the retention period in this format:
`P[<integer>D]T[<integer>H][<integer>M][<integer>S] | P<integer>W`
For example, if you want to enter a retention period of one day, two hours, three minutes, and four seconds, enter the value as `P1DT2H3M4S`.

```
vserver object-store-server user regenerate-keys -vserver svm_name  
-user user -key-time-to-live 0
```

3. Save the access and secret keys. They will be required for access from S3 clients.

System Manager

1. Click **Storage > Storage VMs** and then select the storage VM.
2. In the **Settings** tab, click  in the **S3** tile.
3. In the **Users** tab, verify that there is no access key, or the key has expired for the user.
4. If you need to regenerate the key, click  next to the user, then click **Regenerate Key**.
5. By default, generated keys are valid for an indefinite amount of time. Beginning with 9.14.1, you can modify their retention period, after which the keys automatically expire. Enter the retention period in days, hours, minutes, or seconds.
6. Click **Save**. The key is regenerated. Any change in the key retention period takes effect immediately.
7. Download or save the access key and secret key. They will be required for access from S3 clients.

Create or modify access policy statements

About bucket and object store server policies

User and group access to S3 resources is controlled by bucket and object store server policies. If you have a small number of users or groups, controlling access at the bucket level is probably sufficient, but if you have many users and groups, it is easier to control access at the object store server level.

Modify a bucket policy

You can add access rules to the default bucket policy. The scope of its access control is the containing bucket, so it is most appropriate when there is a single bucket.

Before you begin

An S3-enabled storage VM containing an S3 server and a bucket must already exist.

You must have already created users or groups before granting permissions.

About this task

You can add new statements for new users and groups, or you can modify the attributes of existing statements. For more options, see the `vserver object-store-server bucket policy` man pages.

User and group permissions can be granted when the bucket is created or as needed later. You can also modify the bucket capacity and QoS policy group assignment.

Beginning with ONTAP 9.9.1, if you plan to support AWS client object tagging functionality with the ONTAP S3 server, the actions `GetObjectTagging`, `PutObjectTagging`, and `DeleteObjectTagging` need to be allowed using the bucket or group policies.

The procedure you follow depends on the interface that you use—System Manager or the CLI:

System Manager

Steps

1. Edit the bucket: click **Storage > Buckets**, click the desired bucket, and then click **Edit**.

When adding or modifying permissions, you can specify the following parameters:

- **Principal**: the user or group to whom access is granted.
- **Effect**: allows or denies access to a user or group.
- **Actions**: permissible actions in the bucket for a given user or group.
- **Resources**: paths and names of objects within the bucket for which access is granted or denied.

The defaults **bucketname** and **bucketname/*** grant access to all objects in the bucket. You can also grant access to single objects; for example, **bucketname/*_readme.txt**.

- **Conditions** (optional): expressions that are evaluated when access is attempted. For example, you can specify a list of IP addresses for which access will be allowed or denied.



Beginning with ONTAP 9.14.1, you can specify variables for the bucket policy in the **Resources** field. These variables are placeholders that are replaced with contextual values when the policy is evaluated. For example, If `${aws:username}` is specified as a variable for a policy, then this variable is replaced with the request context username, and the policy action can be performed as configured for that user.

CLI

Steps

1. Add a statement to a bucket policy:

```
vserver object-store-server bucket policy add-statement -vserver svm_name
-bucket bucket_name -effect {allow|deny} -action object_store_actions
-principal user_and_group_names -resource object_store_resources [-sid
text] [-index integer]
```

The following parameters define access permissions:

-effect	The statement may allow or deny access
-action	You can specify * to mean all actions, or a list of one or more of the following: GetObject, PutObject, DeleteObject, ListBucket, GetBucketAcl, GetObjectAcl, ListBucketMultipartUploads, and ListMultipartUploadParts.

-principal	<p>A list of one or more S3 users or groups.</p> <ul style="list-style-type: none"> • A maximum of 10 users or groups can be specified. • If an S3 group is specified, it must be in the form <code>group/group_name</code>. • <code>*</code> can be specified to mean public access; that is, access without an access-key and secret-key. • If no principal is specified, all S3 users in the storage VM are granted access.
-resource	<p>The bucket and any object it contains. The wildcard characters <code>*</code> and <code>?</code> can be used to form a regular expression for specifying a resource. For a resource, you can specify variables in a policy. These are policy variables are placeholders that are replaced with the contextual values when the policy is evaluated.</p>

You can optionally specify a text string as comment with the `-sid` option.

Examples

The following example creates an object store server bucket policy statement for the storage VM `svm1.example.com` and `bucket1` which specifies allowed access to a `readme` folder for object store server user `user1`.

```
cluster1::> vservers object-store-server bucket policy statement create
-vserver svm1.example.com -bucket bucket1 -effect allow -action
GetObject,PutObject,DeleteObject,ListBucket -principal user1 -resource
bucket1/readme/* -sid "fullAccessToReadmeForUser1"
```

The following example creates an object store server bucket policy statement for the storage VM `svm1.example.com` and `bucket1` which specifies allowed access to all objects for object store server group `group1`.

```
cluster1::> vservers object-store-server bucket policy statement create
-vserver svm1.example.com -bucket bucket1 -effect allow -action
GetObject,PutObject,DeleteObject,ListBucket -principal group/group1
-resource bucket1/* -sid "fullAccessForGroup1"
```

Beginning with ONTAP 9.14.1, you can specify variables for a bucket policy. The following example creates a server bucket policy statement for the storage VM `svm1` and `bucket1`, and specifies `${aws:username}` as a variable for a policy resource. When the policy is evaluated, the policy variable is replaced with the request context username, and the policy action can be performed as configured for that user. For example, when the following policy statement is evaluated, `${aws:username}` is replaced with the user performing the S3 operation. If a user `user1` performs the operation, that user is granted access to `bucket1` as `bucket1/user1/*`.

```
cluster1::> object-store-server bucket policy statement create -vserver  
svml -bucket bucket1 -effect allow -action * -principal - -resource  
bucket1,bucket1/${aws:username}/*##
```

Create or modify an object store server policy

You can create policies that can apply to one or more buckets in an object store. Object store server policies can be attached to groups of users, thereby simplifying the management of resource access across multiple buckets.

Before you begin

An S3-enabled SVM containing an S3 server and a bucket must already exist.

About this task

You can enable access policies at the SVM level by specifying a default or custom policy in an object storage server group. The policies do not take effect until they are specified in the group definition.



When you use object storage server policies, you specify principals (that is, users and groups) in the group definition, not in the policy itself.

There are three read-only default policies for access to ONTAP S3 resources:

- FullAccess
- NoS3Access
- ReadOnlyAccess

You can also create new custom policies, then add new statements for new users and groups, or you can modify the attributes of existing statements. For more options, see the `vserver object-store-server policy` [command reference](#).


Beginning with ONTAP 9.9.1, if you plan to support AWS client object tagging functionality with the ONTAP S3 server, the actions `GetObjectTagging`, `PutObjectTagging`, and `DeleteObjectTagging` need to be allowed using the bucket or group policies.

The procedure you follow depends on the interface that you use—System Manager or the CLI:

System Manager

Use System Manager to create or modify an object store server policy

Steps

1. Edit the storage VM: click **Storage > storage VMs**, click the storage VM, click **Settings** and then click  under S3.
2. Add a user: click **Policies**, then click **Add**.
 - a. Enter a policy name and select from a list of groups.
 - b. Select an existing default policy or add a new one.

When adding or modifying a group policy, you can specify the following parameters:

- Group: the groups to whom access is granted.
- Effect: allows or denies access to one or more groups.
- Actions: permissible actions in one or more buckets for a given group.
- Resources: paths and names of objects within one or more buckets for which access is granted or denied.

For example:

- * grants access to all buckets in the storage VM.
 - **bucketname** and **bucketname/*** grant access to all objects in a specific bucket.
 - **bucketname/readme.txt** grants access to an object in a specific bucket.
- c. If desired, add statements to existing policies.

CLI

Use the CLI to create or modify an object store server policy

Steps

1. Create an object storage server policy:

```
vserver object-store-server policy create -vserver svm_name -policy policy_name [-comment text]
```

2. Create a statement for the policy:

```
vserver object-store-server policy statement create -vserver svm_name -policy policy_name -effect {allow|deny} -action object_store_actions -resource object_store_resources [-sid text]
```

The following parameters define access permissions:

-effect	The statement may allow or deny access
---------	--

<code>-action</code>	You can specify <code>*</code> to mean all actions, or a list of one or more of the following: <code>GetObject</code> , <code>PutObject</code> , <code>DeleteObject</code> , <code>ListBucket</code> , <code>GetBucketAcl</code> , <code>GetObjectAcl</code> , <code>ListAllMyBuckets</code> , <code>ListBucketMultipartUploads</code> , and <code>ListMultipartUploadParts</code> .
<code>-resource</code>	The bucket and any object it contains. The wildcard characters <code>*</code> and <code>?</code> can be used to form a regular expression for specifying a resource.

You can optionally specify a text string as comment with the `-sid` option.

By default, new statements are added to the end of the list of statements, which are processed in order. When you add or modify statements later, you have the option to modify the statement's `-index` setting to change the processing order.

Configure S3 access for external directory services

Beginning with ONTAP 9.14.1, services for external directories have been integrated with ONTAP S3 object storage. This integration simplifies user and access management through external directory services.

You can provide user groups belonging to an external directory service with access to your ONTAP object storage environment. Lightweight Directory Access Protocol (LDAP) is an interface for communicating with directory services, such as Active Directory, that provide a database and services for identity and access management (IAM). To provide access, you need to configure LDAP groups in your ONTAP S3 environment. After you have configured access, the group members have permissions to ONTAP S3 buckets. For information about LDAP, see [Overview of using LDAP](#).

You can also configure Active Directory user groups for fast bind mode, so that user credentials can be validated and third-party and open-source S3 applications can be authenticated over LDAP connections.

Before you begin

Ensure the following before configuring LDAP groups and enabling the fast bind mode for group access:

1. An S3-enabled storage VM containing an S3 server has been created. See [Create an SVM for S3](#).
2. A bucket has been created in that storage VM. See [Create a bucket](#).
3. DNS is configured on the storage VM. See [Configure DNS services](#).
4. A self-signed root certification authority (CA) certificate of the LDAP server is installed on the storage VM. See [Install the self-signed root CA certificate on the SVM](#).
5. An LDAP client is configured with TLS enabled on the SVM. See [Create an LDAP client configuration](#) and [Associate the LDAP client configuration with SVMs for information](#).

Configure S3 access for external directory services

1. Specify LDAP as the *name service database* of the SVM for the group and password to LDAP:

```
ns-switch modify -vserver <vserver-name> -database group -sources
files,ldap
ns-switch modify -vserver <vserver-name> -database passwd -sources
files,ldap
```

For more information about this command, see the [vserver services name-service ns-switch modify](#) command.

2. Create an object store bucket policy statement with the `principal` set to the LDAP group to which you want to grant access:

```
object-store-server bucket policy statement create -bucket <bucket-name>
-effect allow -principal nasgroup/<ldap-group-name> -resource <bucket-
name>, <bucket-name>/*
```

Example: The following example creates a bucket policy statement for `buck1`. The policy allows access for the LDAP group `group1` to the resource (bucket and its objects) `buck1`.

```
vserver object-store-server bucket policy add-statement -bucket buck1
-effect allow -action
GetObject,PutObject,DeleteObject,ListBucket,GetBucketAcl,GetObjectAcl,Li
stBucketMultipartUploads,ListMultipartUploadParts,
ListBucketVersions,GetObjectTagging,PutObjectTagging,DeleteObjectTagging
,GetBucketVersioning,PutBucketVersioning -principal nasgroup/group1
-resource buck1, buck1/*
```

3. Verify that a user from the LDAP group `group1` is able to perform S3 operations from the S3 client.

Use LDAP fast bind mode for authentication

1. Specify LDAP as the *name service database* of the SVM for the group and password to LDAP:

```
ns-switch modify -vserver <vserver-name> -database group -sources
files,ldap
ns-switch modify -vserver <vserver-name> -database passwd -sources
files,ldap
```

For more information about this command, see the [vserver services name-service ns-switch modify](#) command.

2. Ensure that an LDAP user accessing the S3 bucket has permissions defined in the bucket-policies. For more information, see [Modify a bucket policy](#).
3. Verify that a user from the LDAP group can perform the following operations:

- a. Configure the access key on the S3 client in this format:

"NTAPFASTBIND" + base64-encode(user-name:password)

Example: "NTAPFASTBIND" + base64-encode(ldapuser:password), which results in
NTAPFASTBINDbGRhcHVzZXI6cGFzc3dvcmQ=



The S3 client might prompt for a secret key. In the absence of a secret key, any password of at least 16 characters can be entered.

- b. Perform basic S3 operations from the S3 client for which the user has permissions.

Enable LDAP or domain users to generate their own S3 access keys

Beginning with ONTAP 9.14.1, as an ONTAP administrator, you can create custom roles and grant them to local or domain groups or Lightweight Directory Access Protocol (LDAP) groups, so that the users belonging to those groups can generate their own access and secret keys for S3 client access.

You have to perform a few configuration steps on your storage VM, so that the custom role can be created and assigned to the user that invokes the API for access key generation.

Before you begin

Ensure the following:

1. An S3-enabled storage VM containing an S3 server has been created. See [Create an SVM for S3](#).
2. A bucket has been created in that storage VM. See [Create a bucket](#).
3. DNS is configured on the storage VM. See [Configure DNS services](#).
4. A self-signed root certification authority (CA) certificate of the LDAP server is installed on the storage VM. See [Install the self-signed root CA certificate on the SVM](#).
5. An LDAP client is configured with TLS enabled on the storage VM. See [Create an LDAP client configuration](#) and .
6. Associate the client configuration with the Vserver. See [Associate the LDAP client configuration with SVMs and vservice services name-service ldap create](#).
7. If you are using a data storage VM, create a management network interface (LIF) and on the VM, and also a service policy for the LIF. See the [network interface create](#) and [network interface service-policy create](#) commands.

Configure users for access key generation

1. Specify LDAP as the *name service database* of the storage VM for the group and password to LDAP:

```
ns-switch modify -vserver <vserver-name> -database group -sources  
files,ldap  
ns-switch modify -vserver <vserver-name> -database passwd -sources  
files,ldap
```

For more information about this command, see the [vserver services name-service ns-switch modify](#) command.

2. Create a custom role with access to S3 user REST API endpoint:

```
security login rest-role create -vserver <vserver-name> -role <custom-role-name> -api "/api/protocols/s3/services/*/users" -access <access-type>
```

In this example, the `s3-role` role is generated for users on the storage VM `svm-1`, to which all access rights, read, create, and update are granted.

```
security login rest-role create -vserver svm-1 -role s3role -api  
"/api/protocols/s3/services/*/users" -access all
```

For more information about this command, see the [security login rest-role create](#) command.

3. Create an LDAP user group with the security login command and add the new custom role for accessing the S3 user REST API endpoint. For more information about this command, see the [security login create](#) command.

```
security login create -user-or-group-name <ldap-group-name> -application  
http -authentication-method nsswitch -role <custom-role-name> -is-ns  
-switch-group yes
```

In this example, the LDAP group `ldap-group-1` is created in `svm-1`, and the custom role `s3role` is added to it for accessing the API endpoint, along with enabling LDAP access in the fast bind mode.

```
security login create -user-or-group-name ldap-group-1 -application http  
-authentication-method nsswitch -role s3role -is-ns-switch-group yes  
-second-authentication-method none -vserver svm-1 -is-ldap-fastbind yes
```

For more information, see [Use LDAP fast bind for nsswitch authentication](#).

Adding the custom role to the domain or LDAP group allows users in that group a limited access to the ONTAP `/api/protocols/s3/services/{svm.uuid}/users` endpoint. By invoking the API, the domain or LDAP group users can generate their own access and secret keys to access the S3 client. They can generate the keys for only themselves and not for other users.

As an S3 or LDAP user, generate your own access keys

Beginning with ONTAP 9.14.1, you can generate your own access and secret keys for accessing S3 clients, if your administrator has granted you the role to generate your own keys. You can generate keys for only yourself by using the following ONTAP REST API endpoint.

HTTP method and endpoint

This REST API call uses the following method and endpoint. For information about the other methods of this endpoint, see the reference [API documentation](#).

HTTP method	Path
POST	/api/protocols/s3/services/{svm.uuid}/users

Curl example

```
curl
--request POST \
--location "https://$FQDN_IP /api/protocols/s3/services/{svm.uuid}/users "
\
--include \
--header "Accept: */*" \
--header "Authorization: Basic $BASIC_AUTH"
--data '{"name": "_name_"}'
```

JSON output example

```
{
  "records": [
    {
      "access_key":
      "Pz3SB54G2B_6dsXQPrA5HrTPcf478qoAW6_Xx6qyqZ948AgZ_7YfCf_9nO87YoZmskxx3cq41
      U2JAH2M3_fs321B4rkzS3a_oC5_8u7D8j_45N8OsBCBPWGD_1d_ccfq",
      "_links": {
        "next": {
          "href": "/api/resourcelink"
        },
        "self": {
          "href": "/api/resourcelink"
        }
      },
      "name": "user-1",
      "secret_key":
      "A20_tDhC_cux2C2BmtL45bXB_a_Q65c_96FsAcOdo14Az8V31jBKDTc0uCL62Bh559gPB8s9r
      rn0868QrF38_1dsV2u1_9H2tSf3qQ5xp9NT259C6z_GiZQ883Qn63X1"
    }
  ],
  "num_records": "1"
}
```

Enable client access to S3 object storage

Enable ONTAP S3 access for remote FabricPool tiering

For ONTAP S3 to be used as a remote FabricPool capacity (cloud) tier, the ONTAP S3 administrator must provide information about the S3 server configuration to the remote ONTAP cluster administrator.

About this task

The following S3 server information is required to configure FabricPool cloud tiers:

- server name (FQDN)
- bucket name
- CA certificate
- access key
- password (secret access key)

In addition, the following networking configuration is required:

- There must be an entry for the remote ONTAP S3 server's hostname in the DNS server configured for the admin SVM, including the S3 server's FQDN name and the IP addresses on its LIFs.
- Intercluster LIFs must be configured on the local cluster, although cluster peering is not required.

See the FabricPool documentation about configuring ONTAP S3 as a cloud tier.

Managing Storage Tiers By Using FabricPool

Enable ONTAP S3 access for local FabricPool tiering

For ONTAP S3 to be used as a local FabricPool capacity tier, you must define an object store based on the bucket you created, and then attach the object store to a performance tier aggregate to create a FabricPool.

Before you begin

You must have the ONTAP S3 server name and a bucket name, and the S3 server must have been created using cluster LIFs (with the `-vserver Cluster` parameter).

About this task

The object-store configuration contains information about the local capacity tier, including the S3 server and bucket names and authentication requirements.

An object-store configuration once created must not be reassociated with a different object-store or bucket. You can create multiple buckets for local tiers, but you cannot create multiple object stores in a single bucket.

A FabricPool license is not required for a local capacity tier.

Steps

1. Create the object store for the local capacity tier:

```
storage aggregate object-store config create -object-store-name store_name
-ipSPACE Cluster -provider-type ONTAP_S3 -server S3_server_name -container
-name bucket_name -access-key access_key -secret-password password
```

- The `-container-name` is the S3 bucket you created.
- The `-access-key` parameter authorizes requests to the ONTAP S3 server.
- The `-secret-password` parameter (secret access key) authenticates requests to the ONTAP S3 server.
- You can set the `-is-certificate-validation-enabled` parameter to `false` to disable certificate checking for ONTAP S3.

```
cluster1::> storage aggregate object-store config create
-object-store-name MyLocalObjStore -ipspace Cluster -provider-type
ONTAP_S3 -server s3.example.com
-container-name bucket1 -access-key myS3key -secret-password myS3pass
```

2. Display and verify the object store configuration information:

```
storage aggregate object-store config show
```

3. Optional: To see how much data in a volume is inactive, follow the steps in [Determining how much data in a volume is inactive by using inactive data reporting](#).

Seeing how much data in a volume is inactive can help you decide which aggregate to use for FabricPool local tiering.

4. Attach the object store to an aggregate:

```
storage aggregate object-store attach -aggregate aggr_name -object-store-name
store_name
```

You can use the `allow-flexgroup true` option to attach aggregates that contain FlexGroup volume constituents.

```
cluster1::> storage aggregate object-store attach
-aggregate aggr1 -object-store-name MyLocalObjStore
```

5. Display the object store information and verify that the attached object store is available:

```
storage aggregate object-store show
```

```
cluster1::> storage aggregate object-store show
```

Aggregate	Object Store Name	Availability State
-----	-----	-----
aggr1	MyLocalObjStore	available

Enable client access from an S3 app

For S3 client apps to access the ONTAP S3 server, the ONTAP S3 administrator must provide configuration information to the S3 user.

Before you begin

The S3 client app must be capable of authenticating with the ONTAP S3 server using the following AWS signature versions:

- Signature Version 4, ONTAP 9.8 and later

- Signature Version 2, ONTAP 9.11.1 and later

Other signature versions are not supported by ONTAP S3.

The ONTAP S3 administrator must have created S3 users and granted them access permissions, as an individual users or as a group member, in the bucket policy or the object storage server policy.

The S3 client app must be capable of resolving the ONTAP S3 server name, which requires that ONTAP S3 administrator provide the S3 server name (FQDN) and IP addresses for the S3 server's LIFs.

About this task

To access an ONTAP S3 bucket, a user on the S3 client app enters information provided by the ONTAP S3 administrator.

Beginning with ONTAP 9.9.1, the ONTAP S3 server supports the following AWS client functionality:

- user-defined object metadata

A set of key-value pairs can be assigned to objects as metadata when they are created using PUT (or POST). When a GET/HEAD operation is performed on the object, the user-defined metadata is returned along with the system metadata.

- object tagging

A separate set of key-value pairs can be assigned as tags for categorizing objects. Unlike metadata, tags are created and read with REST APIs independently of the object, and they implemented when objects are created or any time after.



To enable clients to get and put tagging information, the actions `GetObjectTagging`, `PutObjectTagging`, and `DeleteObjectTagging` need to be allowed using the bucket or group policies.

For more information, see the AWS S3 documentation.

Steps

1. Authenticate the S3 client app with the ONTAP S3 server by entering the S3 server name and the CA certificate.
2. Authenticate a user on the S3 client app by entering the following information:
 - S3 server name (FQDN) and bucket name
 - the user's access key and secret key

Storage service definitions

ONTAP includes predefined storage services that are mapped to corresponding minimum performance factors.

The actual set of storage services available in a cluster or SVM is determined by the type of storage that makes up an aggregate in the SVM.

The following table shows how the minimum performance factors are mapped to the predefined storage services:

Storage service	Expected IOPS (SLA)	Peak IOPS (SLO)	Minimum volume IOPS	Estimated latency	Are expected IOPS enforced?
value	128 per TB	512 per TB	75	17 ms	On AFF: Yes Otherwise: No
performance	2048 per TB	4096 per TB	500	2 ms	Yes
extreme	6144 per TB	12288 per TB	1000	1 ms	Yes

The following table defines the available storage service level for each type of media or node:

Media or node	Available storage service level
Disk	value
Virtual machine disk	value
FlexArray LUN	value
Hybrid	value
Capacity-optimized Flash	value
Solid-state drive (SSD) - non-AFF	value
Performance-optimized Flash - SSD (AFF)	extreme, performance, value

Protect buckets with S3 SnapMirror

S3 SnapMirror overview

Beginning with ONTAP 9.10.1, you can protect buckets in ONTAP S3 object stores using SnapMirror mirroring and backup functionality. Unlike standard SnapMirror, S3 SnapMirror enables mirroring and backups to non-NetApp destinations like AWS S3.

S3 SnapMirror supports active mirrors and backup tiers from ONTAP S3 buckets to the following destinations:

Target	Supports active mirrors and takeover?	Supports backup and restore?
ONTAP S3 <ul style="list-style-type: none"> • buckets in the same SVM • buckets in different SVMs on the same cluster • buckets in SVMs on different clusters 	✓	✓
StorageGRID		✓
AWS S3		✓
Cloud Volumes ONTAP for Azure	✓	✓
Cloud Volumes ONTAP for AWS	✓	✓
Cloud Volumes ONTAP for Google Cloud	✓	✓

You can protect existing buckets on ONTAP S3 servers or you can create new buckets with data protection enabled immediately.

S3 SnapMirror requirements

- ONTAP version
ONTAP 9.10.1 or later must be running on source and destination clusters.
- Licensing
The following license bundles are required on ONTAP source and destination systems:
 - Core Bundle
For ONTAP S3 protocol and storage.
 - Data Protection Bundle
For S3 SnapMirror to target other NetApp object store targets (ONTAP S3, StorageGRID, and Cloud Volumes ONTAP).
 - Data Protection Bundle and Hybrid Cloud Bundle
For S3 SnapMirror to target third-party object stores, including AWS S3.
- ONTAP S3
 - ONTAP S3 servers must be running source and destination SVMs.
 - It is recommended but not required that CA certificates for TLS access are installed on systems that host S3 servers.
 - The CA certificates used to sign the S3 servers' certificates must be installed on the admin storage VM of the clusters that host S3 servers.
 - You can use a self-signed CA certificate or a certificate signed by an external CA vendor.
 - If the source or destination storage VMs are not listening on HTTPS, it is not necessary to install CA certificates.
- Peering (for ONTAP S3 targets)
 - Intercluster LIFs must be configured (for remote ONTAP targets).
 - Source and destination clusters are peered (for remote ONTAP targets).

- Source and destination storage VMs are peered (for all ONTAP targets).
- SnapMirror policy
 - An S3-specific SnapMirror policy is required for all S3 SnapMirror relationships, but you can use the same policy for multiple relationships.
 - You can create your own policy or accept the default **Continuous** policy, which includes the following values:
 - Throttle (upper limit on throughput/bandwidth) - unlimited.
 - Time for recovery point objective: 1 hour (3600 seconds).
- Root user keys

Storage VM root user access keys are required for S3 SnapMirror relationships; ONTAP does not assign them by default. The first time you create an S3 SnapMirror relationship, you must verify that the keys exist on both source and destination storage VMs and regenerate them if they do not. If you need to regenerate them, you must ensure that all clients and all SnapMirror object-store configurations using the access and secret key pair are updated with the new keys.

For information about S3 server configuration, see the following topics:

- [Enable an S3 server on a storage VM](#)
- [About the S3 configuration process](#)

For information about cluster and storage VM peering, see the following topic:

- [Prepare for mirroring and vaulting \(System Manager, steps 1-6\)](#)
- [Cluster and SVM peering \(CLI\)](#)

Supported SnapMirror relationships

S3 SnapMirror supports fan-out and cascade relationships. For an overview, see [Fan-out and cascade data protection deployments](#).

S3 SnapMirror does not support fan-in deployments (data protection relationships between multiple source buckets and a single destination bucket). S3 Snapmirror can support multiple bucket mirrors from multiple clusters to a single secondary cluster, but each source bucket must have its own destination bucket on the secondary cluster.

Control access to S3 buckets

When you create new buckets, you can control access by creating users and groups. For more information, see the following topics:

- [Add S3 users and groups \(System Manager\)](#)
- [Create an S3 user \(CLI\)](#)
- [Create or modify S3 groups \(CLI\)](#)

Mirror and backup protection on a remote cluster

Create a mirror relationship for a new bucket (remote cluster)

When you create new S3 buckets, you can protect them immediately to an S3 SnapMirror

destination on a remote cluster.



About this task

You will need to perform tasks on both source and destination systems.

Before you begin


- Requirements for ONTAP versions, licensing, and S3 server configuration have been completed.
- A peering relationship exists between source and destination clusters, and a peering relationship exists between source and destination storage VMs.
- CA Certificates are needed for the source and destination VMs. You can use self-signed CA certificates or certificates signed by an external CA vendor.

System Manager

1. If this is the first S3 SnapMirror relationship for this storage VM, verify that root user keys exist for both source and destination storage VMs and regenerate them if they do not:
 - a. Click **Storage > Storage VMs** and then select the storage VM.
 - b. In the **Settings** tab, click  in the **S3** tile.
 - c. In the **Users** tab, verify that there is an access key for the root user.
 - d. If there is not, click  next to **root**, then click **Regenerate Key**.
Do not regenerate the key if one already exists.
2. Edit the storage VM to add users, and to add users to groups, in both the source and destination storage VMs:

Click **Storage > storage VMs**, click the storage VM, click **Settings** and then click  under S3.

See [Add S3 users and groups](#) for more information.

3. On the source cluster, create an S3 SnapMirror policy if you don't have an existing one and you don't want to use the default policy:
 - a. Click **Protection > Overview**, and then click **Local Policy Settings**.
 - b. Click  next to **Protection Policies**, then click **Add**.
 - Enter the policy name and description.
 - Select the policy scope, cluster or SVM
 - Select **Continuous** for S3 SnapMirror relationships.
 - Enter your **Throttle** and **Recovery Point Objective** values.
4. Create a bucket with SnapMirror protection:
 - a. Click **Storage > Buckets**, then click **Add**. Verifying permissions is optional but recommended.
 - b. Enter a name, select the storage VM, enter a size, then click **More Options**.
 - c. Under **Permissions**, click **Add**.
 - **Principal** and **Effect** - select values corresponding to your user group settings or accept the defaults.
 - **Actions**- make sure the following values are shown:

```
GetObject,PutObject,DeleteObject,ListBucket,GetBucketAcl,GetObjectAcl,ListBucketMultipartUploads,ListMultipartUploadParts
```

- **Resources** - use the defaults (*bucketname*, *bucketname/**) or other values you need.

See [Manage user access to buckets](#) for more information about these fields.

- d. Under **Protection**, check **Enable SnapMirror (ONTAP or Cloud)**. Then enter the following values:
 - Destination
 - **TARGET: ONTAP System**

- **CLUSTER:** Select the remote cluster.
 - **STORAGE VM:** Select a storage VM on the remote cluster.
 - **S3 SERVER CA CERTIFICATE:** Copy and paste the contents of the *source* certificate.
 - Source
 - **S3 SERVER CA CERTIFICATE:** Copy and paste the contents of the *destination* certificate.
5. Check **Use the same certificate on the destination** if you are using a certificate signed by an external CA vendor.
 6. If you click **Destination Settings**, you can also enter your own values in place of the defaults for bucket name, capacity, and performance service level.
 7. Click **Save**. A new bucket is created in the source storage VM, and it is mirrored to a new bucket that is created the destination storage VM.

Back up locked buckets

Beginning with ONTAP 9.14.1, you can back up locked S3 buckets and restore them as required.

When defining the protection settings for a new or existing bucket, you can enable object locking on destination buckets, provided that the source and destination clusters run ONTAP 9.14.1 or later, and that object locking is enabled on the source bucket. The object locking mode and lock retention tenure of the source bucket become applicable for the replicated objects on the destination bucket. You can also define a different lock retention period for the destination bucket in the **Destination Settings** section. This retention period is also applied to any non-locked objects replicated from the source bucket and S3 interfaces.

For information about how to enable object locking on a bucket, see [Create a bucket](#).

CLI

1. If this is the first S3 SnapMirror relationship for this SVM, verify that root user keys exist for both source and destination SVMs and regenerate them if they do not:

```
vserver object-store-server user show
```

Verify that there is an access key for the root user. If there is not, enter:

```
vserver object-store-server user regenerate-keys -vserver svm_name -user
root
```

Do not regenerate the key if one already exists.

2. Create buckets in both the source and destination SVMs:

```
vserver object-store-server bucket create -vserver svm_name -bucket
bucket_name [-size integer[KB|MB|GB|TB|PB]] [-comment text]
[additional_options]
```

3. Add access rules to the default bucket policies in both the source and destination SVMs:

```
vserver object-store-server bucket policy add-statement -vserver svm_name
-bucket bucket_name -effect {allow|deny} -action object_store_actions
-principal user_and_group_names -resource object_store_resources [-sid
text] [-index integer]
```

Example

```
src_cluster::> vserver object-store-server bucket policy add-  
statement -bucket test-bucket -effect allow -action  
GetObject,PutObject,DeleteObject,ListBucket,GetBucketAcl,GetObjectAc  
l,ListBucketMultipartUploads,ListMultipartUploadParts -principal -  
-resource test-bucket, test-bucket /*
```

4. On the source SVM, create an S3 SnapMirror policy if you don't have an existing one and you don't want to use the default policy:

```
snapmirror policy create -vserver svm_name -policy policy_name -type  
continuous [-rpo integer] [-throttle throttle_type] [-comment text]  
[additional_options]
```

Parameters:

- `type continuous` - the only policy type for S3 SnapMirror relationships (required).
- `-rpo` - specifies the time for recovery point objective, in seconds (optional).
- `-throttle` - specifies the upper limit on throughput/bandwidth, in kilobytes/seconds (optional).

Example

```
src_cluster::> snapmirror policy create -vserver vs0 -type  
continuous -rpo 0 -policy test-policy
```

5. Install CA server certificates on the admin SVMs of the source and destination clusters:

- a. On the source cluster, install the CA certificate that signed the *destination* S3 server certificate:

```
security certificate install -type server-ca -vserver src_admin_svm  
-cert-name dest_server_certificate
```
- b. On the destination cluster, install the CA certificate that signed the *source* S3 server certificate:

```
security certificate install -type server-ca -vserver dest_admin_svm  
-cert-name src_server_certificate
```

If you are using a certificate signed by an external CA vendor, install the same certificate on the source and destination admin SVM.

See the `security certificate install` man page for details.

6. On the source SVM, create an S3 SnapMirror relationship:

```
snapmirror create -source-path src_svm_name:/bucket/bucket_name  
-destination-path dest_peer_svm_name:/bucket/bucket_name, ...} [-policy  
policy_name]
```

You can use a policy you created or accept the default.

Example

```
src_cluster::> snapmirror create -source-path vs0-src:/bucket/test-  
bucket -destination-path vs1-dest:bucket/test-bucket-mirror -policy  
test-policy
```

7. Verify that mirroring is active:

```
snapmirror show -policy-type continuous -fields status
```

Create a mirror relationship for an existing bucket (remote cluster)

You can begin protecting existing S3 buckets at any time; for example, if you upgraded an S3 configuration from a release earlier than ONTAP 9.10.1.

About this task

You need to perform tasks on both the source and destination clusters.




Before you begin

- Requirements for ONTAP versions, licensing, and S3 server configuration have been completed.
- A peering relationship exists between source and destination clusters, and a peering relationship exists between source and destination storage VMs.
- CA Certificates are needed for the source and destination VMs. You can use self-signed CA certificates or certificates signed by an external CA vendor.



Steps

You can create a mirror relationship using System Manager or the ONTAP CLI.

System Manager

1. If this is the first S3 SnapMirror relationship for this storage VM, verify that root user keys exist for both source and destination storage VMs and regenerate them if they do not:
 - a. Select **Storage > Storage VMs** and then select the storage VM.
 - b. In the **Settings** tab, click  in the **S3** tile.
 - c. In the **Users** tab, verify that there is an access key for the root user.
 - d. If there is not, click  next to **root**, then click **Regenerate Key**.
Do not regenerate the key if one already exists.
2. Verify that user and group access is correct in both the source and destination storage VMs: Select **Storage > storage VMs**, then select the storage VM, then **Settings**. Finally, select  under **S3**.

See [Add S3 users and groups](#) for more information.

3. On the source cluster, create an S3 SnapMirror policy if you don't have an existing one and you don't want to use the default policy:
 - a. Select **Protection > Overview**, and then click **Local Policy Settings**.
 - b. Select  next to **Protection Policies**, then click **Add**.
 - c. Enter the policy name and description.
 - d. Select the policy scope, cluster or SVM
 - e. Select **Continuous** for S3 SnapMirror relationships.
 - f. Enter your **Throttle** and **Recovery Point Objective** values.
4. Verify that the bucket access policy of the existing bucket still meets your needs:
 - a. Click **Storage > Buckets** and then select the bucket you want to protect.
 - b. In the **Permissions** tab, click  **Edit**, then click **Add** under **Permissions**.
 - **Principal and Effect**: select values corresponding to your user group settings, or accept the defaults.
 - **Actions**: make sure the following values are shown:

```
GetObject, PutObject, DeleteObject, ListBucket, GetBucketAcl, GetObjectAcl, ListBucketMultipartUploads, ListMultipartUploadParts
```

- **Resources**: use the defaults (*bucketname*, *bucketname/**) or other values you need.

See [Manage user access to buckets](#) for more information about these fields.

5. Protect an existing bucket with S3 SnapMirror protection:
 - a. Click **Storage > Buckets** and then select the bucket you want to protect..
 - b. Click **Protect** and enter the following values:
 - Destination
 - **TARGET**: ONTAP System

- **CLUSTER:** Select the remote cluster.
 - **STORAGE VM:** Select a storage VM on the remote cluster.
 - **S3 SERVER CA CERTIFICATE:** Copy and paste the contents of the *source* certificate.
 - **Source**
 - **S3 SERVER CA CERTIFICATE:** Copy and paste the contents of the *destination* certificate.
6. Check **Use the same certificate on the destination** if you are using a certificate signed by an external CA vendor.
 7. If you click **Destination Settings**, you can also enter your own values in place of the defaults for bucket name, capacity, and performance service level.
 8. Click **Save**. The existing bucket is mirrored to a new bucket in the destination storage VM.

Back up locked buckets

Beginning with ONTAP 9.14.1, you can back up locked S3 buckets and restore them as required.

When defining the protection settings for a new or existing bucket, you can enable object locking on destination buckets, provided that the source and destination clusters run ONTAP 9.14.1 or later, and that object locking is enabled on the source bucket. The object locking mode and lock retention tenure of the source bucket become applicable for the replicated objects on the destination bucket. You can also define a different lock retention period for the destination bucket in the **Destination Settings** section. This retention period is also applied to any non-locked objects replicated from the source bucket and S3 interfaces.

For information about how to enable object locking on a bucket, see [Create a bucket](#).

CLI

1. If this is the first S3 SnapMirror relationship for this SVM, verify that root user keys exist for both source and destination SVMs and regenerate them if they do not:

```
vserver object-store-server user show
```

Verify that there is an access key for the root user. If there is not, enter:

```
vserver object-store-server user regenerate-keys -vserver svm_name -user root
```

Do not regenerate the key if one already exists.

2. Create a bucket on the destination SVM to be the mirror target:

```
vserver object-store-server bucket create -vserver svm_name -bucket dest_bucket_name [-size integer[KB|MB|GB|TB|PB]] [-comment text] [additional_options]
```

3. Verify that the access rules of the default bucket policies are correct in both the source and destination SVMs:

```
vserver object-store-server bucket policy add-statement -vserver svm_name -bucket bucket_name -effect {allow|deny} -action object_store_actions -principal user_and_group_names -resource object_store_resources [-sid text] [-index integer]
```

Example

```
src_cluster::> vserver object-store-server bucket policy add-  
statement -bucket test-bucket -effect allow -action  
GetObject,PutObject,DeleteObject,ListBucket,GetBucketAcl,GetObjectAc  
l,ListBucketMultipartUploads,ListMultipartUploadParts -principal -  
-resource test-bucket, test-bucket /*
```

4. On the source SVM, create an S3 SnapMirror policy if you don't have an existing one and you don't want to use the default policy:

```
snapmirror policy create -vserver svm_name -policy policy_name -type  
continuous [-rpo integer] [-throttle throttle_type] [-comment text]  
[additional_options]
```

Parameters:

- `continuous` – the only policy type for S3 SnapMirror relationships (required).
- `-rpo` – specifies the time for recovery point objective, in seconds (optional).
- `-throttle` – specifies the upper limit on throughput/bandwidth, in kilobytes/seconds (optional).

Example

```
src_cluster::> snapmirror policy create -vserver vs0 -type  
continuous -rpo 0 -policy test-policy
```

5. Install CA certificates on the admin SVMs of source and destination clusters:

- a. On the source cluster, install the CA certificate that signed the *destination* S3 server certificate:

```
security certificate install -type server-ca -vserver src_admin_svm  
-cert-name dest_server_certificate
```
- b. On the destination cluster, install the CA certificate that signed the *source* S3 server certificate:

```
security certificate install -type server-ca -vserver dest_admin_svm  
-cert-name src_server_certificate
```

If you are using a certificate signed by an external CA vendor, install the same certificate on the source and destination admin SVM.

See the `security certificate install` man page for details.

6. On the source SVM, create an S3 SnapMirror relationship:

```
snapmirror create -source-path src_svm_name:/bucket/bucket_name  
-destination-path dest_peer_svm_name:/bucket/bucket_name, ...} [-policy  
policy_name]
```

You can use a policy you created or accept the default.

Example

```
src_cluster::> snapmirror create -source-path vs0:/bucket/test-bucket -destination-path vs1:/bucket/test-bucket-mirror -policy test-policy
```

7. Verify that mirroring is active:

```
snapmirror show -policy-type continuous -fields status
```

Takeover and serve data from the destination bucket (remote cluster)

If the data in a source bucket becomes unavailable, you can break the SnapMirror relationship to make the destination bucket writable and begin serving data.

About this task


When a takeover operation is performed, source bucket is converted to read-only and original destination bucket is converted to read-write, thereby reversing the S3 SnapMirror relationship.

When the disabled source bucket is available again, S3 SnapMirror automatically resynchronizes the contents of the two buckets. It is not necessary to explicitly resynchronize the relationship, as is required for volume SnapMirror deployments.

The takeover operation must be initiated from the remote cluster.

System Manager

Failover from the unavailable bucket and begin serving data:

1. Click **Protection > Relationships**, then select **S3 SnapMirror**.
2. Click , select **Failover**, then click **Failover**.

CLI

1. Initiate a failover operation for the destination bucket:

```
snapmirror failover start -destination-path svm_name:/bucket/bucket_name
```
2. Verify the status of the failover operation:

```
snapmirror show -fields status
```

Example

```
dest_cluster::> snapmirror failover start -destination-path dest_svm1:/bucket/test-bucket-mirror
```

Restore a bucket from the destination storage VM (remote cluster)

If data in a source bucket is lost or corrupted, you can repopulate your data by restoring objects from a destination bucket.

About this task


You can restore the destination bucket to an existing bucket or a new bucket. The target bucket for the restore operation must be larger than the destination bucket's logical used space.

If you use an existing bucket, it must be empty when starting a restore operation. Restore does not "roll back" a bucket in time; rather, it populates an empty bucket with its previous contents.

The restore operation must be initiated from the remote cluster.

System Manager

Restore the backed up data:

1. Click **Protection > Relationships**, then select **S3 SnapMirror**.
2. Click  and then select **Restore**.
3. Under **Source**, select **Existing Bucket** (the default) or **New Bucket**.
 - To restore to an **Existing Bucket** (the default), complete these actions:
 - Select the cluster and storage VM to search for the existing bucket.
 - Select the existing bucket.
 - Copy and paste the contents of the *destination* S3 server CA certificate.
 - To restore to a **New Bucket**, enter the following values:
 - The cluster and storage VM to host the new bucket.
 - The new bucket's name, capacity, and performance service level.
See [Storage service levels](#) for more information.
 - The contents of the *destination* S3 server CA certificate.
4. Under **Destination**, copy and paste the contents of the *source* S3 server CA certificate.
5. Click **Protection > Relationships** to monitor the restore progress.

Restore locked buckets

Beginning with ONTAP 9.14.1, you can back up locked buckets and restore them as needed.

You can restore an object-locked bucket to a new or existing bucket. You can select an object-locked bucket as the destination in the following scenarios:

- **Restore to a new bucket:** When object locking is enabled, a bucket can be restored by creating a bucket that also has object locking enabled. When you restore a locked bucket, the object locking mode and retention period of the original bucket are replicated. You can also define a different lock retention period for the new bucket. This retention period is applied to non-locked objects from other sources.
- **Restore to an existing bucket:** An object-locked bucket can be restored to an existing bucket, as long as versioning and a similar object-locking mode are enabled on the existing bucket. The retention tenure of the original bucket is maintained.
- **Restore non-locked bucket:** Even if object locking is not enabled on a bucket, you can restore it to a bucket that has object locking enabled and is on the source cluster. When you restore the bucket, all the non-locked objects become locked, and the retention mode and tenure of the destination bucket become applicable to them.

CLI

1. Create the new destination bucket for restore. For more information, see [Create a backup relationship for a new bucket \(cloud target\)](#).
2. Initiate a restore operation for the destination bucket:

```
snapmirror restore -source-path svm_name:/bucket/bucket_name -destination  
-path svm_name:/bucket/bucket_name
```

Example

```
dest_cluster::> snapmirror restore -source-path src_vs1:/bucket/test-  
bucket -destination-path dest_vs1:/bucket/test-bucket-mirror
```

Mirror and backup protection on the local cluster




Create a mirror relationship for a new bucket (local cluster)


When you create new S3 buckets, you can protect them immediately to an S3 SnapMirror destination on the same cluster. You can mirror data to a bucket in a different storage VM or the same storage VM as the source.

Before you begin

- Requirements for ONTAP versions, licensing, and S3 server configuration have been completed.
- A peering relationship exists between source and destination storage VMs.
- CA Certificates are needed for the source and destination VMs. You can use self-signed CA certificates or certificates signed by an external CA vendor.

System Manager

1. If this is the first S3 SnapMirror relationship for this storage VM, verify that root user keys exist for both source and destination storage VMs and regenerate them if they do not:
 - a. Click **Storage > Storage VMs** and then select the storage VM.
 - b. In the **Settings** tab, click  in the S3 tile.
 - c. In the **Users** tab, verify that there is an access key for the root user
 - d. If there is not, click  next to **root**, then click **Regenerate Key**.
Do not regenerate the key if one already exists.
2. Edit the storage VM to add users, and to add users to groups, in both the source and destination storage VMs:
Click **Storage > storage VMs**, click the storage VM, click **Settings** and then click  under S3.

See [Add S3 users and groups](#) for more information.
3. Create an S3 SnapMirror policy if you don't have an existing one and you don't want to use the default policy:
 - a. Click **Protection > Overview**, and then click **Local Policy Settings**.
 - b. Click  next to **Protection Policies**, then click **Add**.
 - Enter the policy name and description.
 - Select the policy scope, cluster or SVM
 - Select **Continuous** for S3 SnapMirror relationships.
 - Enter your **Throttle** and **Recovery Point Objective** values.
4. Create a bucket with SnapMirror protection:
 - a. Click **Storage > Buckets** then click **Add**.
 - b. Enter a name, select the storage VM, enter a size, then click **More Options**.
 - c. Under **Permissions**, click **Add**. Verifying permissions is optional but recommended.
 - **Principal** and **Effect** - select values corresponding to your user group settings, or accept the defaults.
 - **Actions** - make sure the following values are shown:

```
GetObject,PutObject,DeleteObject,ListBucket,GetBucketAcl,GetObjectAcl,ListBucketMultipartUploads,ListMultipartUploadParts
```
 - **Resources** - use the defaults (`bucketname`, `bucketname/*`) or other values you need
See [Manage user access to buckets](#) for more information about these fields.
 - d. Under **Protection**, check **Enable SnapMirror (ONTAP or Cloud)**. Then enter the following values:
 - **Destination**
 - **TARGET**: ONTAP System
 - **CLUSTER**: Select the local cluster.

- **STORAGE VM:** Select a storage VM on the local cluster.
 - **S3 SERVER CA CERTIFICATE:** Copy and paste the contents of the source certificate.
- Source
- **S3 SERVER CA CERTIFICATE:** Copy and paste the contents of the destination certificate.
5. Check **Use the same certificate on the destination** if you are using a certificate signed by an external CA vendor.
 6. If you click **Destination Settings**, you can also enter your own values in place of the defaults for bucket name, capacity, and performance service level.
 7. Click **Save**. A new bucket is created in the source storage VM, and it is mirrored to a new bucket that is created the destination storage VM.

Back up locked buckets

Beginning with ONTAP 9.14.1, you can back up locked S3 buckets and restore them as required.

When defining the protection settings for a new or existing bucket, you can enable object locking on destination buckets, provided that the source and destination clusters run ONTAP 9.14.1 or later, and that object locking is enabled on the source bucket. The object locking mode and lock retention tenure of the source bucket become applicable for the replicated objects on the destination bucket. You can also define a different lock retention period for the destination bucket in the **Destination Settings** section. This retention period is also applied to any non-locked objects replicated from the source bucket and S3 interfaces.

For information about how to enable object locking on a bucket, see [Create a bucket](#).

CLI

1. If this is the first S3 SnapMirror relationship for this SVM, verify that root user keys exist for both source and destination SVMs and regenerate them if they do not:

```
vserver object-store-server user show
```

Verify that there is an access key for the root user. If there is not, enter:

```
vserver object-store-server user regenerate-keys -vserver svm_name -user root
```

Do not regenerate the key if one already exists.

2. Create buckets in both the source and destination SVMs:

```
vserver object-store-server bucket create -vserver svm_name -bucket
bucket_name [-size integer[KB|MB|GB|TB|PB]] [-comment text]
[additional_options]
```

3. Add access rules to the default bucket policies in both the source and destination SVMs:

```
vserver object-store-server bucket policy add-statement -vserver svm_name
-bucket bucket_name -effect {allow|deny} -action object_store_actions
-principal user_and_group_names -resource object_store_resources [-sid
text] [-index integer]
```

Example

```
src_cluster::> vserver object-store-server bucket policy add-  
statement -bucket test-bucket -effect allow -action  
GetObject,PutObject,DeleteObject,ListBucket,GetBucketAcl,GetObjectAc  
l,ListBucketMultipartUploads,ListMultipartUploadParts -principal -  
-resource test-bucket, test-bucket /*
```

4. Create an S3 SnapMirror policy if you don't have an existing one and you don't want to use the default policy:

```
snapmirror policy create -vserver svm_name -policy policy_name -type  
continuous [-rpo integer] [-throttle throttle_type] [-comment text]  
[additional_options]
```

Parameters:

- `continuous` – the only policy type for S3 SnapMirror relationships (required).
- `-rpo` – specifies the time for recovery point objective, in seconds (optional).
- `-throttle` – specifies the upper limit on throughput/bandwidth, in kilobytes/seconds (optional).

Example

```
src_cluster::> snapmirror policy create -vserver vs0 -type  
continuous -rpo 0 -policy test-policy
```

5. Install CA server certificates on the admin SVM:

- Install the CA certificate that signed the *source* S3 server's certificate on the admin SVM:

```
security certificate install -type server-ca -vserver admin_svm -cert  
-name src_server_certificate
```
- Install the CA certificate that signed the *destination* S3 server's certificate on the admin SVM:

```
security certificate install -type server-ca -vserver admin_svm -cert  
-name dest_server_certificate
```

If you are using a certificate signed by an external CA vendor, you only need to install this certificate on the admin SVM.

See the `security certificate install` man page for details.

6. Create an S3 SnapMirror relationship:

```
snapmirror create -source-path src_svm_name:/bucket/bucket_name  
-destination-path dest_peer_svm_name:/bucket/bucket_name, ...} [-policy  
policy_name]
```

You can use a policy you created or accept the default.

Example

```
src_cluster::> snapmirror create -source-path vs0-src:/bucket/test-  
bucket -destination-path vs1-dest:/vs1/bucket/test-bucket-mirror  
-policy test-policy
```

7. Verify that mirroring is active:

```
snapmirror show -policy-type continuous -fields status
```




Create a mirror relationship for an existing bucket (local cluster)

You can begin protecting existing S3 buckets on the same cluster at any time; for example, if you upgraded an S3 configuration from a release earlier than ONTAP 9.10.1. You can mirror data to a bucket in a different storage VM or the same storage VM as the source.



Before you begin

- Requirements for ONTAP versions, licensing, and S3 server configuration have been completed.
- A peering relationship exists between source and destination storage VMs.
- CA Certificates are needed for the source and destination VMs. You can use self-signed CA certificates or certificates signed by an external CA vendor.

System Manager

1. If this is the first S3 SnapMirror relationship for this storage VM, verify that root user keys exist for both source and destination storage VMs and regenerate them if they do not:
 - a. Click **Storage > Storage VMs** and then select the storage VM.
 - b. In the **Settings** tab, click  in the **S3** tile.
 - c. In the **Users** tab, verify that there is an access key for the root user.
 - d. If there is not, click  next to **root**, then click **Regenerate Key**.
Do not regenerate the key if one already exists
2. Verify that user and group access is correct in both the source and destination storage VMs:
 - Click **Storage > storage VMs**, click the storage VM, click **Settings** and then click  under S3.

See [Add S3 users and groups](#) for more information.

3. Create an S3 SnapMirror policy if you don't have an existing one and you don't want to use the default policy:
 - a. Click **Protection > Overview**, and then click **Local Policy Setting**.
 - b. Click  next to **Protection Policies**, then click **Add**.
 - Enter the policy name and description.
 - Select the policy scope, cluster or SVM
 - Select **Continuous** for S3 Snapmirror relationships.
 - Enter your **Throttle** and **Recovery Point Objective** values.
4. Verify that the bucket access policy of the existing bucket continues to meet your needs:
 - a. Click **Storage > Buckets** and then select the bucket you want to protect.
 - b. In the **Permissions** tab, click  **Edit**, then click **Add** under **Permissions**.
 - **Principal** and **Effect** - select values corresponding to your user group settings, or accept the defaults.
 - **Actions** - make sure the following values are shown:

```
GetObject, PutObject, DeleteObject, ListBucket, GetBucketAcl, GetObjectAcl, ListBucketMultipartUploads, ListMultipartUploadParts
```

- **Resources** - use the defaults (*bucketname*, *bucketname/**) or other values you need.

See [Manage user access to buckets](#) for more information about these fields.

5. Protect an existing bucket with S3 SnapMirror:
 - a. Click **Storage > Buckets** and then select the bucket you want to protect.
 - b. Click **Protect** and enter the following values:
 - **Destination**
 - **TARGET**: ONTAP System
 - **CLUSTER**: Select the local cluster.

- **STORAGE VM:** Select the same or a different storage VM.
 - **S3 SERVER CA CERTIFICATE:** Copy and paste the contents of the *source* certificate.
- Source
 - **S3 SERVER CA CERTIFICATE:** Copy and paste the contents of the *destination* certificate.
6. Check **Use the same certificate on the destination** if you are using a certificate signed by an external CA vendor.
 7. If you click **Destination Settings**, you can also enter your own values in place of the defaults for bucket name, capacity, and performance service level.
 8. Click **Save**. The existing bucket is mirrored to a new bucket in the destination storage VM.

Back up locked buckets

Beginning with ONTAP 9.14.1, you can back up locked S3 buckets and restore them as required.

When defining the protection settings for a new or existing bucket, you can enable object locking on destination buckets, provided that the source and destination clusters run ONTAP 9.14.1 or later, and that object locking is enabled on the source bucket. The object locking mode and lock retention tenure of the source bucket become applicable for the replicated objects on the destination bucket. You can also define a different lock retention period for the destination bucket in the **Destination Settings** section. This retention period is also applied to any non-locked objects replicated from the source bucket and S3 interfaces.

For information about how to enable object locking on a bucket, see [Create a bucket](#).

CLI

1. If this is the first S3 SnapMirror relationship for this SVM, verify that root user keys exist for both source and destination SVMs and regenerate them if they do not:

```
vserver object-store-server user show
```

Verify that there is an access key for the root user. If there is not, enter:

```
vserver object-store-server user regenerate-keys -vserver svm_name -user root
```

Do not regenerate the key if one already exists.

2. Create a bucket on the destination SVM to be the mirror target:

```
vserver object-store-server bucket create -vserver svm_name -bucket dest_bucket_name [-size integer[KB|MB|GB|TB|PB]] [-comment text] [additional_options]
```

3. Verify that the access rules to the default bucket policies are correct in both the source and destination SVMs:

```
vserver object-store-server bucket policy add-statement -vserver svm_name -bucket bucket_name -effect {allow|deny} -action object_store_actions -principal user_and_group_names -resource object_store_resources [-sid text] [-index integer]`
```

Example

```
clusterA::> vsserver object-store-server bucket policy add-statement
-bucket test-bucket -effect allow -action
GetObject,PutObject,DeleteObject,ListBucket,GetBucketAcl,GetObjectAcl,
ListBucketMultipartUploads,ListMultipartUploadParts -principal -
-resource test-bucket, test-bucket /*
```

4. Create an S3 SnapMirror policy if you don't have an existing one and you don't want to use the default policy:

```
snapmirror policy create -vsserver svm_name -policy policy_name -type
continuous [-rpo _integer] [-throttle throttle_type] [-comment text]
[additional_options]
```

Parameters:

- `continuous` – the only policy type for S3 SnapMirror relationships (required).
- `-rpo` – specifies the time for recovery point objective, in seconds (optional).
- `-throttle` – specifies the upper limit on throughput/bandwidth, in kilobytes/seconds (optional).

Example

```
clusterA::> snapmirror policy create -vsserver vs0 -type
continuous -rpo 0 -policy test-policy
```

5. Install CA server certificates on the admin SVM:

- a. Install the CA certificate that signed the *source* S3 server's certificate on the admin SVM:

```
security certificate install -type server-ca -vsserver admin_svm -cert
-name src_server_certificate
```
- b. Install the CA certificate that signed the *destination* S3 server's certificate on the admin SVM:

```
security certificate install -type server-ca -vsserver admin_svm -cert
-name dest_server_certificate
```

If you are using a certificate signed by an external CA vendor, you only need to install this certificate on the admin SVM.

See the `security certificate install` man page for details.

6. Create an S3 SnapMirror relationship:

```
snapmirror create -source-path src_svm_name:/bucket/bucket_name
-destination-path dest_peer_svm_name:/bucket/bucket_name, ...} [-policy
policy_name]
```

You can use a policy you created or accept the default.

Example

```
src_cluster::> snapmirror create -source-path vs0-src:/bucket/test-bucket -destination-path vs1-dest:/bucket/test-bucket-mirror -policy test-policy
```

7. Verify that mirroring is active:

```
snapmirror show -policy-type continuous -fields status
```

Takeover and serve data from the destination bucket (local cluster)

If the data in a source bucket becomes unavailable, you can break the SnapMirror relationship to make the destination bucket writable and begin serving data.

About this task


When a takeover operation is performed, source bucket is converted to read-only and original destination bucket is converted to read-write, thereby reversing the S3 SnapMirror relationship.

When the disabled source bucket is available again, S3 SnapMirror automatically resynchronizes the contents of the two buckets. You don't need to explicitly resynchronize the relationship, as is required for standard volume SnapMirror deployments.

If the destination bucket is on a remote cluster, the takeover operation must be initiated from the remote cluster.

System Manager

Failover from the unavailable bucket and begin serving data:

1. Click **Protection > Relationships**, then select **S3 SnapMirror**.
2. Click , select **Failover**, then click **Failover**.

CLI

1. Initiate a failover operation for the destination bucket:

```
snapmirror failover start -destination-path svm_name:/bucket/bucket_name
```
2. Verify the status of the failover operation:

```
snapmirror show -fields status
```

Example

```
clusterA::> snapmirror failover start -destination-path vs1:/bucket/test-bucket-mirror
```

Restore a bucket from the destination storage VM (local cluster)

When data in a source bucket is lost or corrupted, you can repopulate your data by restoring objects from a destination bucket.

About this task


You can restore the destination bucket to an existing bucket or a new bucket. The target bucket for the restore operation must be larger than the destination bucket's logical used space.

If you use an existing bucket, it must be empty when starting a restore operation. Restore does not "roll back" a bucket in time; rather, it populates an empty bucket with its previous contents.

The restore operation must be initiated from the local cluster.

System Manager

Restore the back-up data:

1. Click **Protection > Relationships**, then select the bucket.
2. Click  and then select **Restore**.
3. Under **Source**, select **Existing Bucket** (the default) or **New Bucket**.
 - To restore to an **Existing Bucket** (the default), complete these actions:
 - Select the cluster and storage VM to search for the existing bucket.
 - Select the existing bucket.
4. Copy and paste the contents of the destination S3 server CA certificate.
 - To restore to a **New Bucket**, enter the following values:
 - The cluster and storage VM to host the new bucket.
 - The new bucket's name, capacity, and performance service level.
See [Storage service levels](#) for more information.
 - The contents of the destination S3 server CA certificate.
5. Under **Destination**, copy and paste the contents of the source S3 server CA certificate.
6. Click **Protection > Relationships** to monitor the restore progress.

Restore locked buckets

Beginning with ONTAP 9.14.1, you can back up locked buckets and restore them as needed.

You can restore an object-locked bucket to a new or existing bucket. You can select an object-locked bucket as the destination in the following scenarios:

- **Restore to a new bucket:** When object locking is enabled, a bucket can be restored by creating a bucket that also has object locking enabled. When you restore a locked bucket, the object locking mode and retention period of the original bucket are replicated. You can also define a different lock retention period for the new bucket. This retention period is applied to non-locked objects from other sources.
- **Restore to an existing bucket:** An object-locked bucket can be restored to an existing bucket, as long as versioning and a similar object-locking mode are enabled on the existing bucket. The retention tenure of the original bucket is maintained.
- **Restore non-locked bucket:** Even if object locking is not enabled on a bucket, you can restore it to a bucket that has object locking enabled and is on the source cluster. When you restore the bucket, all the non-locked objects become locked, and the retention mode and tenure of the destination bucket become applicable to them.

CLI

1. If you are restoring objects to a new bucket, create the new bucket. For more information, see [Create a backup relationship for a new bucket \(cloud target\)](#).
2. Initiate a restore operation for the destination bucket:

```
snapmirror restore -source-path svm_name:/bucket/bucket_name -destination  
-path svm_name:/bucket/bucket_name
```

Example

```
clusterA::> snapmirror restore -source-path vs0:/bucket/test-bucket  
-destination-path vs1:/bucket/test-bucket-mirror
```

Backup protection with cloud targets

Requirements for cloud target relationships

Make sure that your source and target environments meet the requirements for S3 SnapMirror backup protection to cloud targets.

You must have valid account credentials with the object store provider to access the data bucket.

Intercluster network interfaces and an IPspace should be configured on the cluster before the cluster can connect to a cloud object store. You should create intercluster network interfaces on each node to seamlessly transfer data from the local storage to the cloud object store.

For StorageGRID targets, you need to know the following information:

- server name, expressed as a fully-qualified domain name (FQDN) or IP address
- bucket name; the bucket must already exist
- access key
- secret key

In addition, the CA certificate used to sign the StorageGRID server certificate needs to be installed on the ONTAP S3 cluster's admin storage VM using the `security certificate install` command. For more information, see [Installing a CA certificate](#) if you use StorageGRID.

For AWS S3 targets, you need to know the following information:

- server name, expressed as a fully-qualified domain name (FQDN) or IP address
- bucket name; the bucket must already exist
- access key
- secret key

The DNS server for the ONTAP cluster's admin storage VM must be able to resolve FQDNs (if used) to IP addresses.

Create a backup relationship for a new bucket (cloud target)


When you create new S3 buckets, you can back them up immediately to an S3 SnapMirror target bucket on an object store provider, which can be a StorageGRID system or an Amazon S3 deployment.

Before you begin


- You have valid account credentials and configuration information for the object store provider.

- Intercluster network interfaces and an IPspace have been configured on the source system.
- • The DNS configuration for the source storage VM must be able to resolve the target's FQDN.

System Manager

1. Edit the storage VM to add users, and to add users to groups:
 - a. Click **Storage > storage VMs**, click the storage VM, click **Settings** and then click  under **S3**.

See [Add S3 users and groups](#) for more information.

2. Add a Cloud Object Store on the source system:
 - a. Click **Protection > Overview**, then select **Cloud Object Stores**.
 - b. Click **Add**, then select **Amazon S3** or **StorageGRID**.
 - c. Enter the following values:
 - Cloud object store name
 - URL style (path or virtual-hosted)
 - storage VM (enabled for S3)
 - Object store server name (FQDN)
 - Object store certificate
 - Access key
 - Secret key
 - Container (bucket) name
3. Create an S3 SnapMirror policy if you don't have an existing one and you don't want to use the default policy:
 - a. Click **Protection > Overview**, and then click **Local Policy Settings**.
 - b. Click  next to **Protection Policies**, then click **Add**.
 - Enter the policy name and description.
 - Select the policy scope, cluster or SVM
 - Select **Continuous** for S3 SnapMirror relationships.
 - Enter your **Throttle** and **Recovery Point Objective** values.
4. Create a bucket with SnapMirror protection:
 - a. Click **Storage > Buckets**, then click **Add**.
 - b. Enter a name, select the storage VM, enter a size, then click **More Options**.
 - c. Under **Permissions**, click **Add**. Verifying permissions is optional but recommended.
 - **Principal** and **Effect** - select values corresponding to your user group settings or accept the defaults.
 - **Actions** - make sure the following values are shown:

```
`GetObject,PutObject,DeleteObject,ListBucket,GetBucketAcl,GetObjectAcl,ListBucketMultipartUploads,ListMultipartUploadParts`
```

- **Resources** - use the defaults `_(bucketname, bucketname/*)` or other values you need.

See [Manage user access to buckets](#) for more information about these fields.

- d. Under **Protection**, check **Enable SnapMirror (ONTAP or Cloud)**, select **Cloud Storage**, then select the **Cloud Object Store**.

When you click **Save**, a new bucket is created in the source storage VM, and it is backed up to the cloud object store.

CLI

1. If this is the first S3 SnapMirror relationship for this SVM, verify that root user keys exist for both source and destination SVMs and regenerate them if they do not:

```
vserver object-store-server user show
```

Confirm that there is an access key for the root user. If there is not, enter:

```
vserver object-store-server user regenerate-keys -vserver svm_name -user root
```

Do not regenerate the key if one already exists.

2. Create a bucket in the source SVM:

```
vserver object-store-server bucket create -vserver svm_name -bucket bucket_name [-size integer[KB|MB|GB|TB|PB]] [-comment text] [additional_options]
```

3. Add access rules to the default bucket policy:

```
vserver object-store-server bucket policy add-statement -vserver svm_name -bucket bucket_name -effect {allow|deny} -action object_store_actions -principal user_and_group_names -resource object_store_resources [-sid text] [-index integer]
```

Example

```
clusterA::> vserver object-store-server bucket policy add-statement -bucket test-bucket -effect allow -action GetObject,PutObject,DeleteObject,ListBucket,GetBucketAcl,GetObjectAcl,ListBucketMultipartUploads,ListMultipartUploadParts -principal - -resource test-bucket, test-bucket /*
```

4. Create an S3 SnapMirror policy if you don't have an existing one and you don't want to use the default policy:

```
snapmirror policy create -vserver svm_name -policy policy_name -type continuous [-rpo integer] [-throttle throttle_type] [-comment text] [additional_options]
```

Parameters:

- * `type continuous` – the only policy type for S3 SnapMirror relationships (required).
- * `-rpo` – specifies the time for recovery point objective, in seconds (optional).
- * `-throttle` – specifies the upper limit on throughput/bandwidth, in kilobytes/seconds (optional).

Example

```
clusterA::> snapmirror policy create -vserver vs0 -type continuous -rpo 0 -policy test-policy
```

5. If the target is a StorageGRID system, install the StorageGRID CA server certificate on the admin SVM of the source cluster:

```
security certificate install -type server-ca -vserver src_admin_svm -cert  
-name storage_grid_server_certificate
```

See the `security certificate install` man page for details.

6. Define the S3 SnapMirror destination object store:

```
snapmirror object-store config create -vserver svm_name -object-store-name  
target_store_name -usage data -provider-type {AWS_S3|SGWS} -server  
target_FQDN -container-name remote_bucket_name -is-ssl-enabled true -port  
port_number -access-key target_access_key -secret-password  
target_secret_key
```

Parameters:

- * `-object-store-name` – the name of the object store target on the local ONTAP system.
- * `-usage` – use data for this workflow.
- * `-provider-type` – AWS_S3 and SGWS (StorageGRID) targets are supported.
- * `-server` – the target server's FQDN or IP address.
- * `-is-ssl-enabled` –enabling SSL is optional but recommended.

See the `snapmirror object-store config create` man page for details.

Example

```
src_cluster::> snapmirror object-store config create -vserver vs0  
-object-store-name sgws-store -usage data -provider-type SGWS  
-server sgws.example.com -container-name target-test-bucket -is-ssl  
-enabled true -port 443 -access-key abc123 -secret-password xyz890
```

7. Create an S3 SnapMirror relationship:

```
snapmirror create -source-path svm_name:/bucket/bucket_name -destination  
-path object_store_name:/objstore -policy policy_name
```

Parameters:

- * `-destination-path` - the object store name you created in the previous step and the fixed value `objstore`.

You can use a policy you created or accept the default.

Example

```
src_cluster::> snapmirror create -source-path vs0:/bucket/test-  
bucket -destination-path sgws-store:/objstore -policy test-policy
```

8. Verify that mirroring is active:

```
snapmirror show -policy-type continuous -fields status
```


Create a backup relationship for an existing bucket (cloud target)

You can begin backing up existing S3 buckets at any time; for example, if you upgraded an S3 configuration from a release earlier than ONTAP 9.10.1.

Before you begin


- You have valid account credentials and configuration information for the object store provider.
- Intercluster network interfaces and an IPspace have been configured on the source system.
- The DNS configuration for the source storage VM must be able to resolve the target's FQDN.

System Manager

1. Verify that the users and groups are correctly defined:
Click **Storage > storage VMs**, click the storage VM, click **Settings** and then click  under S3.

See [Add S3 users and groups](#) for more information.


2. Create an S3 SnapMirror policy if you don't have an existing one and you don't want to use the default policy:

- a. Click **Protection > Overview**, and then click **Local Policy Settings**.
- b. Click  next to **Protection Policies**, then click **Add**.
- c. Enter the policy name and description.
- d. Select the policy scope, cluster or SVM
- e. Select **Continuous** for S3 SnapMirror relationships.
- f. Enter your **Throttle** and **Recovery Point Objective values**.

3. Add a Cloud Object Store on the source system:

- a. Click **Protection > Overview**, then select **Cloud Object Store**.
- b. Click **Add**, then select **Amazon S3** or **Others** for StorageGRID Webscale.
- c. Enter the following values:
 - Cloud object store name
 - URL style (path or virtual-hosted)
 - storage VM (enabled for S3)
 - Object store server name (FQDN)
 - Object store certificate
 - Access key
 - Secret key
 - Container (bucket) name

4. Verify that the bucket access policy of the existing bucket still meets your needs:

- a. Click **Storage > Buckets** and then select the bucket you want to protect.
- b. In the **Permissions** tab, click  **Edit**, then click **Add** under **Permissions**.
 - **Principal** and **Effect** - select values corresponding to your user group settings or accept the defaults.
 - **Actions** - make sure the following values are shown:
`GetObject, PutObject, DeleteObject, ListBucket, GetBucketAcl, GetObjectAcl, ListBucketMultipartUploads, ListMultipartUploadParts`
 - **Resources** - use the defaults (`bucketname, bucketname/*`) or other values you need.

See [Manage user access to buckets](#) for more information about these fields.

5. Back up the bucket using S3 SnapMirror:

- a. Click **Storage > Buckets** and then select the bucket you want to back up.
- b. Click **Protect**, select **Cloud Storage** under **Target**, then select the **Cloud Object Store**.

When you click **Save**, the existing bucket is backed up to the cloud object store.

CLI

1. Verify that the access rules in the default bucket policy are correct:

```
vserver object-store-server bucket policy add-statement -vserver svm_name
-bucket bucket_name -effect {allow|deny} -action object_store_actions
-principal user_and_group_names -resource object_store_resources [-sid
text] [-index integer]
```

Example

```
clusterA::> vserver object-store-server bucket policy add-statement
-bucket test-bucket -effect allow -action
GetObject,PutObject,DeleteObject,ListBucket,GetBucketAcl,GetObjectAcl,
ListBucketMultipartUploads,ListMultipartUploadParts -principal -
-resource test-bucket, test-bucket /*
```

2. Create an S3 SnapMirror policy if you don't have an existing one and you don't want to use the default policy:

```
snapmirror policy create -vserver svm_name -policy policy_name -type
continuous [-rpo integer] [-throttle throttle_type] [-comment text]
[additional_options]
```

Parameters:

- * `type continuous` – the only policy type for S3 SnapMirror relationships (required).
- * `-rpo` – specifies the time for recovery point objective, in seconds (optional).
- * `-throttle` – specifies the upper limit on throughput/bandwidth, in kilobytes/seconds (optional).

Example

```
clusterA::> snapmirror policy create -vserver vs0 -type continuous
-rpo 0 -policy test-policy
```

3. If the target is a StorageGRID system, install the StorageGRID CA certificate on the admin SVM of the source cluster:

```
security certificate install -type server-ca -vserver src_admin_svm -cert
-name storage_grid_server_certificate
```

See the `security certificate install` man page for details.

4. Define the S3 SnapMirror destination object store:

```
snapmirror object-store config create -vserver svm_name -object-store-name
target_store_name -usage data -provider-type {AWS_S3|SGWS} -server
target_FQDN -container-name remote_bucket_name -is-ssl-enabled true -port
port_number -access-key target_access_key -secret-password
target_secret_key
```

Parameters:

- * `-object-store-name` – the name of the object store target on the local ONTAP system.
- * `-usage` – use data for this workflow.

- * `-provider-type` – AWS_S3 and SGWS (StorageGRID) targets are supported.
- * `-server` – the target server's FQDN or IP address.
- * `-is-ssl-enabled` –enabling SSL is optional but recommended.

See the `snapmirror object-store config create` man page for details.

Example

```
src_cluster::> snapmirror object-store config create -vserver vs0
-object-store-name sgws-store -usage data -provider-type SGWS
-server sgws.example.com -container-name target-test-bucket -is-ssl
-enabled true -port 443 -access-key abc123 -secret-password xyz890
```

5. Create an S3 SnapMirror relationship:

```
snapmirror create -source-path svm_name:/bucket/bucket_name -destination
-path object_store_name:/objstore -policy policy_name
```

Parameters:

- * `-destination-path` - the object store name you created in the previous step and the fixed value `objstore`.

You can use a policy you created or accept the default.

Example

```
src_cluster::> snapmirror create -source-path vs0:/bucket/buck-ebp
-destination-path sgws-store:/objstore -policy test-policy
```

6. Verify that mirroring is active:

```
snapmirror show -policy-type continuous -fields status
```

Restore a bucket from a cloud target

When data in a source bucket is lost or corrupted, you can repopulate your data by restoring from a destination bucket.


About this task

You can restore the destination bucket to an existing bucket or a new bucket. The target bucket for the restore operation must be larger than the destination bucket's logical used space.

If you use an existing bucket, it must be empty when starting a restore operation. Restore does not "roll back" a bucket in time; rather, it populates an empty bucket with its previous contents.

System Manager

Restore the back-up data:

1. Click **Protection > Relationships**, then select **S3 SnapMirror**.
2. Click  and then select **Restore**.
3. Under **Source**, select **Existing Bucket** (the default) or **New Bucket**.
 - To restore to an **Existing Bucket** (the default), complete these actions:
 - Select the cluster and storage VM to search for the existing bucket.
 - Select the existing bucket.
 - Copy and paste the contents of the *destination* S3 server CA certificate.
 - To restore to a **New Bucket**, enter the following values:
 - The cluster and storage VM to host the new bucket.
 - The new bucket's name, capacity, and performance service level.
See [Storage service levels](#) for more information.
 - The contents of the destination S3 server CA certificate.
4. Under **Destination**, copy and paste the contents of the *source* S3 server CA certificate.
5. Click **Protection > Relationships** to monitor the restore progress.

CLI procedure

1. Create the new destination bucket for restore. For more information, see [Create a backup relationship for a bucket \(cloud target\)](#).
2. Initiate a restore operation for the destination bucket:

```
snapmirror restore -source-path object_store_name:/objstore -destination  
-path svm_name:/bucket/bucket_name
```

Example

The following example restores a destination bucket to an existing bucket.


```
clusterA::> snapmirror restore -source-path sgws.store:/objstore  
-destination-path vs0:/bucket/test-bucket
```

Modify a mirror policy

You might want to modify an S3 mirror policy; for example, if you want to adjust the RPO and throttle values.

System Manager

If you want to adjust these values, you can edit an existing protection policy.

1. Click **Protection > Relationships**, and then select the protection policy for the relationship you want to modify.
2. Click  next to the policy name, then click **Edit**.

CLI

Modify an S3 SnapMirror policy:

```
snapmirror policy modify -vserver svm_name -policy policy_name [-rpo integer]
[-throttle throttle_type] [-comment text]
```

Parameters:

- `-rpo` – specifies the time for recovery point objective, in seconds.
- `-throttle` – specifies the upper limit on throughput/bandwidth, in kilobytes/seconds.

Example

```
clusterA::> snapmirror policy modify -vserver vs0 -policy test-policy
-rpo 60
```

Audit S3 events

Audit S3 events

Beginning with ONTAP 9.10.1, you can audit data and management events in ONTAP S3 environments. S3 audit functionality is similar to existing NAS auditing capabilities, and S3 and NAS auditing can coexist in a cluster.

When you create and enable an S3 auditing configuration on an SVM, S3 events are recorded in a log file. The you can specify the following events to be logged:

- Object access (data) events
GetObject, PutObject, and DeleteObject
- Management events
PutBucket and DeleteBucket

The log format is JavaScript Object Notation (JSON).

The combined limit for S3 and NFS auditing configurations is 50 SVMs per cluster.

The following license bundle is required:

- Core Bundle, for ONTAP S3 protocol and storage

For more information, see [How the ONTAP auditing process works](#).

Guaranteed auditing

By default, S3 and NAS auditing is guaranteed. ONTAP guarantees that all auditable bucket access events are recorded, even if a node is unavailable. A requested bucket operation cannot be completed until the audit record for that operation is saved to the staging volume on persistent storage. If audit records cannot be committed in the staging files, either because of insufficient space or because of other issues, client operations are denied.

Space requirements for auditing

In the ONTAP auditing system, audit records are initially stored in binary staging files on individual nodes. Periodically, they are consolidated and converted to user-readable event logs, which are stored in the audit event log directory for the SVM.

The staging files are stored in a dedicated staging volume, which is created by ONTAP when the auditing configuration is created. There is one staging volume per aggregate.

You must plan for sufficient available space in the auditing configuration:

- For the staging volumes in aggregates that contain audited buckets.
- For the volume containing the directory where converted event logs are stored.

You can control the number of event logs, and hence the available space in the volume, using one of two methods when creating the S3 auditing configuration:

- A numerical limit; the `-rotate-limit` parameter controls the minimum number of audit files that must be preserved.
- A time limit; the `-retention-duration` parameter controls the maximum period that files can be preserved.

In both parameters, once that configured is exceeded, older audit files can be deleted to make room for newer ones. For both parameters, the value is 0, indicating that all files must be maintained. In order to ensure sufficient space, it is therefore a best practice to set one of the parameters to a non-zero value.

Because of guaranteed auditing, if the space available for audit data runs out before the rotation limit, newer audit data cannot be created, resulting in failure to clients accessing data. Therefore, the choice of this value and of the space allocated to auditing must be chosen carefully, and you must respond to warnings about available space from the auditing system.

For more information, see [Basic auditing concepts](#).

Plan an S3 auditing configuration

You must specify a number of parameters for the S3 auditing configuration or accept the defaults. In particular, you should consider which log rotation parameters will help ensure adequate free space.

See the `vserver object-store-server audit create` man page for syntax details.

General parameters

There are two required parameters that you must specify when you create the auditing configuration. There are also three optional parameters that you can specify.

Type of information	Option	Required
<i>SVM name</i> Name of the SVM on which to create the auditing configuration. The SVM must already exist and be enabled for S3.	<code>-verserver <i>svm_name</i></code>	Yes
<i>Log destination path</i> Specifies where the converted audit logs are stored. The path must already exist on the SVM. The path can be up to 864 characters in length and must have read-write permissions. If the path is not valid, the audit configuration command fails.	<code>-destination <i>text</i></code>	Yes
<i>Categories of events to audit</i> The following event categories can be audited: <ul style="list-style-type: none">• data GetObject, PutObject, and DeleteObject events• management PutBucket and DeleteBucket events The default is to audit data events only.	<code>-events {data management}, ...</code>	No

You can enter one of the following parameters to control the number of audit log files. If no value is entered, all log files are retained.

Type of information	Option	Required
<i>Log files rotation limit</i> Determines how many audit log files to retain before rotating the oldest log file out. For example, if you enter a value of 5, the last five log files are retained. A value of 0 indicates that all the log files are retained. The default value is 0.	<code>-rotate-limit <i>integer</i></code>	No

<p><i>Log files duration limit</i></p> <p>Determines how long a log file can be retained before being deleted. For example, if you enter a value of 5d0h0m, logs more than 5 days old are deleted.</p> <p>A value of 0 indicates that all the log files are retained. The default value is 0.</p>	<p>-retention duration <i>integer_time</i></p>	<p>No</p>
---	--	-----------

Parameters for audit log rotation

You can rotate audit logs based on size or schedule. The default is to rotate audit logs based on size.

Rotate logs based on log size

If you want to use the default log rotation method and the default log size, you do not need to configure any specific parameters for log rotation. The default log size is 100 MB.

If you do not want to use the default log size, you can configure the `-rotate-size` parameter to specify a custom log size.

If you want to reset the rotation based on a log size alone, use the following command to unset the `-rotate-schedule-minute` parameter:

```
vserver audit modify -vserver svm_name -destination / -rotate-schedule-minute -
```

Rotate logs based on a schedule

If you choose to rotate the audit logs based on a schedule, you can schedule log rotation by using the time-based rotation parameters in any combination.

- If you use time-based rotation, the `-rotate-schedule-minute` parameter is mandatory.
- All other time-based rotation parameters are optional.
 - `-rotate-schedule-month`
 - `-rotate-schedule-dayofweek`
 - `-rotate-schedule-day`
 - `-rotate-schedule-hour`
- The rotation schedule is calculated by using all the time-related values.
For example, if you specify only the `-rotate-schedule-minute` parameter, the audit log files are rotated based on the minutes specified on all days of the week, during all hours on all months of the year.
- If you specify only one or two time-based rotation parameters (for example, `-rotate-schedule-month` and `-rotate-schedule-minutes`), the log files are rotated based on the minute values that you specified on all days of the week, during all hours, but only during the specified months.

For example, you can specify that the audit log is to be rotated during the months January, March, and August on all Mondays, Wednesdays, and Saturdays at 10:30 a.m.

- If you specify values for both `-rotate-schedule-dayofweek` and `-rotate-schedule-day`, they are considered independently.

For example, if you specify `-rotate-schedule-dayofweek` as Friday and `-rotate-schedule-day` as 13, then the audit logs would be rotated on every Friday and on the 13th day of the specified month, not just on every Friday the 13th.

- If you want to reset the rotation based on a schedule alone, use the following command to unset the `-rotate-size` parameter:

```
vserver audit modify -vserver svm_name -destination / -rotate-size -
```

Rotate logs based on log size and schedule

You can choose to rotate the log files based on log size and a schedule by setting both the `-rotate-size` parameter and the time-based rotation parameters in any combination. For example: if `-rotate-size` is set to 10 MB and `-rotate-schedule-minute` is set to 15, the log files rotate when the log file size reaches 10 MB or on the 15th minute of every hour (whichever event occurs first).

Create and enable an S3 auditing configuration

To implement S3 auditing, you first create a persistent object store auditing configuration on an S3-enabled SVM, then enable the configuration.

What you'll need

- An S3-enabled SVM.
- Sufficient space for staging volumes in the aggregate.

About this task

An auditing configuration is required for each SVM that contains S3 buckets that you wish to audit. You can enable S3 auditing on new or existing S3 servers. Auditing configurations persist in an S3 environment until removed by the **vserver object-store-server audit delete** command.

The S3 auditing configuration applies to all buckets in the SVM that you select for auditing. An audit-enabled SVM can contain audited and un-audited buckets.

It is recommended that you configure S3 auditing for automatic log rotation, determined by log size or a schedule. If you don't configure automatic log rotation, all log files are retained by default. You can also rotate S3 log files manually using the **vserver object-store-server audit rotate-log** command.

If the SVM is an SVM disaster recovery source, the destination path cannot be on the root volume.

Procedure

1. Create the auditing configuration to rotate audit logs based on log size or a schedule.

If you want to rotate audit logs by...	Enter...
Log size	<pre>vserver object-store-server audit create -vserver <i>svm_name</i> -destination <i>path</i> [[-events] {data management}, ...] [[-rotate-limit <i>integer</i>] [-retention-duration [<i>integer_d</i>] [<i>integer_h</i>] [<i>integer_m</i>] [<i>integers</i>]]] [-rotate-size {<i>integer</i>[KB MB GB TB PB]}]</pre>

If you want to rotate audit logs by...	Enter...
A schedule	<pre>vserver object-store-server audit create -vserver svm_name -destination path [[-events] {data management}, ...] [[-rotate-limit integer] [- retention-duration [integerd][integerh] [integerm][integers]]] [-rotate-schedule-month chron_month] [-rotate-schedule-dayofweek chron_dayofweek] [- rotate-schedule-day chron_dayofmonth] [-rotate- schedule-hour chron_hour] -rotate-schedule-minute chron_minute</pre> <p>The <code>-rotate-schedule-minute</code> parameter is required if you are configuring time-based audit log rotation.</p>

2. Enable S3 auditing:

```
vserver object-store-server audit enable -vserver svm_name
```

Examples

The following example creates an auditing configuration that audits all S3 events (the default) using size-based rotation. The logs are stored in the `/audit_log` directory. The log file size limit is 200 MB. The logs are rotated when they reach 200 MB in size.

```
cluster1::> vserver audit create -vserver vs1 -destination /audit_log -rotate
-size 200MB
```

The following example creates an auditing configuration that audits all S3 events (the default) using size-based rotation. The log file size limit is 100 MB (the default), and the logs are retained for 5 days before being deleted.

```
cluster1::> vserver audit create -vserver vs1 -destination /audit_log -retention
-duration 5d0h0m
```

The following example creates an auditing configuration that audits S3 management events, and central access policy staging events using time-based rotation. The audit logs are rotated monthly, at 12:30 p.m. on all days of the week. The log rotation limit is 5.

```
cluster1::> vserver audit create -vserver vs1 -destination /audit_log -events
management -rotate-schedule-month all -rotate-schedule-dayofweek all -rotate
-schedule-hour 12 -rotate-schedule-minute 30 -rotate-limit 5
```

Select buckets for S3 auditing

You must specify which buckets to audit in an audit-enabled SVM.

What you'll need

- An SVM enabled for S3 auditing.

About this task

S3 auditing configurations are enabled on a per-SVM basis, but you must select the buckets in SVMs that are enabled for audit. If you add buckets to the SVM and you want the new buckets to be audited, you must select them with this procedure. You can also have non-audited buckets in an SVM enabled for S3 auditing.

Auditing configurations persist for buckets until removed by the `vserver object-store-server audit object-select delete` command.

Procedure

Select a bucket for S3 auditing:

```
vserver object-store-server audit event-selector create -vserver svm_name -bucket bucket_name [[-access] {read-only|write-only|all}] [[-permission] {allow-only|deny-only|all}]
```

- `-access` - specifies the type of event access to be audited: `read-only`, `write-only` or `all` (default is `all`).
- `-permission` - specifies the type of event permission to be audited: `allow-only`, `deny-only` or `all` (default is `all`).

Example

The following example creates a bucket auditing configuration that only logs allowed events with read-only access:

```
cluster1::> vserver object-store-server audit event-selector create -vserver vs1 -bucket test-bucket -access read-only -permission allow-only
```

Modify an S3 auditing configuration

You can modify the auditing parameters of individual buckets or the auditing configuration of all buckets selected for audit in the SVM.

Table 1. Procedure

If you want to modify the audit configuration for...	Enter...
Individual buckets	<code>vserver object-store-server audit event-selector modify -vserver svm_name [-bucket bucket_name] [parameters to modify]</code>
All buckets in the SVM	<code>vserver object-store-server audit modify -vserver svm_name [parameters to modify]</code>

Examples

The following example modifies an individual bucket auditing configuration to audit only write-only access events:

```
cluster1::> vserver object-store-server audit event-selector modify -vserver vs1 -bucket test-bucket -access write-only
```

The following example modifies the auditing configuration of all buckets in the SVM to change the log size limit to 10MB and to retain 3 log files before rotating.

```
cluster1::> vservers object-store-server audit modify -vservers vs1 -rotate
-size 10MB -rotate-limit 3
```

Show S3 auditing configurations

After completing the auditing configuration, you can verify that auditing is configured properly and is enabled. You can also display information about all object store auditing configurations in the cluster.

About this task

You can display information about bucket and SVM auditing configurations.

- Buckets – use the `vservers object-store-server audit event-selector show` command

Without any parameters, the command displays the following information about buckets in all SVMs in the cluster with object store auditing configurations:

- SVM name
- Bucket name
- Access and permission values

- SVMs – use the `vservers object-store-server audit show` command

Without any parameters, the command displays the following information about all SVMs in the cluster with object store auditing configurations:

- SVM name
- Audit state
- Target directory

You can specify the `-fields` parameter to specify which audit configuration information to display.

Procedure

Show information about S3 auditing configurations:

If you want to modify the configuration for...	Enter...
Buckets	<code>vservers object-store-server audit event-selector show [-vservers svm_name] [parameters]</code>
SVMs	<code>vservers object-store-server audit show [-vservers svm_name] [parameters]</code>

Examples

The following example displays information for a single bucket:

```
cluster1::> vserver object-store-server audit event-selector show -vserver
vs1 -bucket test-bucket
```

Vserver	Bucket	Access	Permission
-----	-----	-----	-----
vs1	bucket1	read-only	allow-only

The following example displays information for all buckets on an SVM:

```
cluster1::> vserver object-store-server audit event-selector show -vserver
vs1
```

Vserver	:vs1
Bucket	:test-bucket
Access	:all
Permission	:all

The following example displays the name, audit state, event types, log format, and target directory for all SVMs.

```
cluster1::> vserver object-store-server audit show
```

Vserver	State	Event Types	Log Format	Target Directory
-----	-----	-----	-----	-----
vs1	false	data	json	/audit_log

The following example displays the SVM names and details about the audit log for all SVMs.

```
cluster1::> vserver object-store-server audit show -log-save-details
```

Vserver	Rotation File Size	Rotation Schedule	Rotation Limit
-----	-----	-----	-----
vs1	100MB	-	0

The following example displays in list form all audit configuration information about all SVMs.

```
cluster1::> vserver object-store-server audit show -instance
```

```
          Vserver: vs1
      Auditing state: true
    Log Destination Path: /audit_log
Categories of Events to Audit: data
          Log Format: json
      Log File Size Limit: 100MB
    Log Rotation Schedule: Month: -
Log Rotation Schedule: Day of Week: -
      Log Rotation Schedule: Day: -
      Log Rotation Schedule: Hour: -
    Log Rotation Schedule: Minute: -
          Rotation Schedules: -
      Log Files Rotation Limit: 0
      Log Retention Time: 0s
```

Copyright information

Copyright © 2024 NetApp, Inc. All Rights Reserved. Printed in the U.S. No part of this document covered by copyright may be reproduced in any form or by any means—graphic, electronic, or mechanical, including photocopying, recording, taping, or storage in an electronic retrieval system—without prior written permission of the copyright owner.

Software derived from copyrighted NetApp material is subject to the following license and disclaimer:

THIS SOFTWARE IS PROVIDED BY NETAPP “AS IS” AND WITHOUT ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE, WHICH ARE HEREBY DISCLAIMED. IN NO EVENT SHALL NETAPP BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

NetApp reserves the right to change any products described herein at any time, and without notice. NetApp assumes no responsibility or liability arising from the use of products described herein, except as expressly agreed to in writing by NetApp. The use or purchase of this product does not convey a license under any patent rights, trademark rights, or any other intellectual property rights of NetApp.

The product described in this manual may be protected by one or more U.S. patents, foreign patents, or pending applications.

LIMITED RIGHTS LEGEND: Use, duplication, or disclosure by the government is subject to restrictions as set forth in subparagraph (b)(3) of the Rights in Technical Data -Noncommercial Items at DFARS 252.227-7013 (FEB 2014) and FAR 52.227-19 (DEC 2007).

Data contained herein pertains to a commercial product and/or commercial service (as defined in FAR 2.101) and is proprietary to NetApp, Inc. All NetApp technical data and computer software provided under this Agreement is commercial in nature and developed solely at private expense. The U.S. Government has a non-exclusive, non-transferrable, nonsublicensable, worldwide, limited irrevocable license to use the Data only in connection with and in support of the U.S. Government contract under which the Data was delivered. Except as provided herein, the Data may not be used, disclosed, reproduced, modified, performed, or displayed without the prior written approval of NetApp, Inc. United States Government license rights for the Department of Defense are limited to those rights identified in DFARS clause 252.227-7015(b) (FEB 2014).

Trademark information

NETAPP, the NETAPP logo, and the marks listed at <http://www.netapp.com/TM> are trademarks of NetApp, Inc. Other company and product names may be trademarks of their respective owners.