Provide S3 client access to NAS data

ONTAP 9

NetApp
April 17, 2023

Table of Contents

Provide S3 client access to NAS data

S3 multiprotocol overview

Beginning with ONTAP 9.12.1, you can enable clients running the S3 protocol to access the same data that are being served to clients that use the NFS and SMB protocols without reformatting. This capability allows NAS data to continue to be served to NAS clients, while presenting object data to S3 clients running S3 applications (such as data mining and artificial intelligence).

S3 multiprotocol functionality addresses two use cases:

1. Access to existing NAS data using S3 clients

   If your existing data was created using traditional NAS clients (NFS or SMB) and is located on NAS volumes (FlexVol or FlexGroup volumes), you can now use analytical tools on S3 clients to access this data.

2. Backend storage for modern clients capable of performing I/O using both NAS and S3 protocols

   You can now provide integrated access for applications such Spark and Kafka that can read and write the same data using both NAS and S3 protocols.

How S3 multiprotocol works

ONTAP multiprotocol allows you to present the same data set as a file hierarchy or as objects in a bucket. To do so, ONTAP creates “S3 NAS buckets” that allow S3 clients to create, read, delete, and enumerate files in NAS storage using S3 object requests. This mapping conforms to the NAS security configuration, observing file and directory access permissions as well as writing to the security audit trail as necessary.

This mapping is accomplished by presenting a specified NAS directory hierarchy as an S3 bucket. Each file in the directory hierarchy is represented as an S3 object whose name is relative from the mapped directory downwards, with directory boundaries represented by the slash character ('/').

Normal ONTAP-defined S3 users can access this storage, as governed by the bucket policies defined for the bucket that maps to the NAS directory. For this to be possible, mappings must be defined between the S3 users and SMB/NFS users. The credentials of the SMB/NFS user will be used for the NAS permissions checking and included in any audit records resulting from these accesses.

When created by SMB or NFS clients, a file is immediately placed in a directory, and therefore visible to clients, before any data is written to it. S3 clients expect different semantics, in which the new object is not visible in the namespace until all its data has been written. This mapping of S3 to NAS storage creates files using S3 semantics, keeping the files invisible externally until the S3 creation command completes.

Data protection for S3 NAS buckets

S3 NAS “buckets” are simply mappings of NAS data for S3 clients, they are not standard S3 buckets. Therefore, there is no need to protect S3 NAS buckets using NetApp S3 SnapMirror functionality. Instead, you can replicate source SVMs containing S3 NAS buckets using SVM DR, a standard SnapMirror data protection relationship with destination SVMs. SVM DR is the only supported SnapMirror replication method with S3 multiprotocol. SnapMirror Synchronous is not supported.

Learn about SnapMirror SVM replication.

Auditing for S3 NAS buckets

Because S3 NAS buckets are not conventional S3 buckets, S3 audit cannot be configured to audit access on them. Learn more about S3 audit.

Nonetheless, the NAS files and directories that are mapped in S3 NAS buckets can be audited for access events using conventional ONTAP audit procedures. S3 operations can therefore trigger NAS audit events, with the following exceptions:

- If S3 client access is denied by the S3 policy configuration (group or bucket policy), NAS audit for the event is not initiated. This is because S3 permissions are checked before SVM audit checks can be made.
- If the target file of an S3 Get request is 0 size, 0 content is returned to the Get request and the Read access is not logged.
- If the target file of an S3 Get request is in a folder for which the user has no traverse permission, the access attempt fails and the event is not logged.

Learn about auditing NAS events on SVMs.

S3 and NAS interoperability

ONTAP S3 NAS buckets support standard NAS and S3 functionality except as listed here.

NAS functionality not currently supported by S3 NAS buckets

FabricPool capacity tier

S3 NAS buckets cannot configured as a capacity tier for FabricPool.

S3 functionality not currently supported by S3 NAS buckets

AWS user metadata

- Key-values pairs received as part of S3 user-metadata are not stored on disk along with object data in the current release.
- Request headers with the prefix "x-amz-meta" are ignored.

AWS Tags

- On PUT object and Multipart Initiate requests, headers with the prefix "x-amz-tagging" are ignored.
- Requests to update tags on an existing file (i.e. a Put, Get, and Delete requests with the ?tagging query-string) are rejected with an error.

Versioning

It is not possible to specify versioning in the bucket mapping configuration.

- Requests that include non-null version specifications (the versionId=xyz query-string) receive error responses.
- Requests to affect the versioning state of a bucket are rejected with errors.

Multipart operations

The following operations are not supported:

- AbortMultipartUpload
- CompleteMultipartUpload
- CreateMultipartUpload
- ListMultipartUpload

NAS data requirements for S3 client access

It is important to understand that there are some inherent incompatibilities when mapping NAS files and directories for S3 access. It might be necessary to adjust NAS file hierarchies before serving them using S3 NAS buckets.

An S3 NAS bucket provides S3 access to a NAS directory by mapping that directory using S3 bucket syntax, and the files in the directory tree are viewed as objects. The object names are the slash-delimited pathnames of the files relative to the directory specified in the S3 bucket configuration.

This mapping imposes some requirements when files and directories are served using S3 NAS buckets:

- S3 names are limited to 1024 bytes, so files with longer pathnames are not accessible using S3.
- File and directory names are limited to 255 characters, so an object name cannot have more than 255 consecutive non-slash (‘/’) characters
- An SMB pathname that is delimited by backslash (‘\’) characters will appear to s3 as an object name containing forward-slash (‘/’) characters instead.
- Some pairs of legal S3 object names cannot coexist in the mapped NAS directory tree. For example, the legal S3 object names “part1/part2” and “part1/part2/part3” map to files that cannot simultaneously exist in the NAS directory tree, as “part1/part2” is a file in the first name and a directory in the other.
  - If “part1/part2” is an existing file, an S3 creation of “part1/part2/part3” will fail.
  - If “part1/part2/part3” is an existing file, an S3 creation or deletion of “part1/part2” will fail.
  - An S3 object creation that matches the name of an existing object replaces the pre-existing object (in unversioned buckets); that holds in NAS but requires an exact match. The examples above will not cause removal of the existing object because while the names collide, they do not match.

While an object store is designed to support a very large number of arbitrary names, a NAS directory structure can experience performance problems if a very large number of names are placed in one directory. In particular, names with no slash (‘/’) characters in them will all be placed into the root directory of the NAS mapping. Applications that make extensive use of names that are not “NAS-friendly” would be better hosted on an actual object store bucket rather than a NAS mapping.

Enable S3 protocol access to NAS data

Enabling S3 protocol access consists of ensuring that a NAS-enabled SVM meets the same requirements as an S3-enabled server, including adding an object store server, and verifying networking and authentication requirements.

For new ONTAP installations, it is recommended that you enable S3 protocol access to an SVM after configuring it to serve NAS data to clients. To learn about NAS protocol configuration, see:

- NFS configuration

- SMB configuration

Before you begin

The following must be configured before enabling the S3 protocol:

- The S3 protocol and the desired NAS protocols – NFS, SMB, or both – are licensed.
- An SVM is configured for the desired NAS protocols.
- NFS and/or SMB servers exist.
- DNS and any other required services are configured.
- NAS data is being exported or shared to client systems.

About this task

A Certificate Authority (CA) certificate is required to enable HTTPS traffic from S3 clients to the S3-enabled SVM. CA certificates from three sources can be used:

- A new ONTAP self-signed certificate on the SVM.
- An existing ONTAP self-signed certificate on the SVM.
- A third-party certificate.

You can use the same data LIFs for the S3/NAS bucket that you use for serving NAS data. If specific IP addresses are required, see Create data LIFs. An S3 service data policy is required to enable S3 data traffic on LIFs; you can modify the SVM's existing service policy to include S3.

When you create the S3 object server, 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.

System Manager

1. Enable S3 on a storage VM with NAS protocols configured.
   a. Click **Storage > Storage VMs**, select a NAS-ready storage VM, click Settings, and then click ⚙ under S3.
   b. Select the certificate type. Whether you select system-generated certificate or one of your own, it will be required for client access.
   c. 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 the S3 protocol is allowed on the SVM:
   `vserver show -fields allowed-protocols`
2. Record the public key certificate for this SVM.
   If a new ONTAP self-signed certificate is needed, see Create and install a CA certificate on the SVM.
3. Update the service data policy
   a. Display the service data policy for the SVM
      `network interface service-policy show -vserver svm_name`
   b. Add the `data-core` and `data-s3-server services` if they are not present.
      `network interface service-policy add-service -vserver svm_name -policy policy_name -services data-core,data-s3-server`
4. Verify that the data LIFs on the SVM meet your requirements:
   `network interface show -vserver svm_name`
5. Create the S3 server:
   `vserver object-store-server create -vserver svm_name -object-store-server s3_server_fqdn -certificate-name ca_cert_name -comment text [additional_options]`

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

- 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 proper 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, all the request and responses are sent over the network in clear text.

6. Verify that S3 is configured as desired:
   `vserver object-store-server show`

Example
The following 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
```

Create S3 NAS bucket

An S3 NAS buckets is a mapping between an S3 bucket name and a NAS path. S3 NAS buckets allow you to provide S3 access to any part of an SVM namespace having existing volumes and directory structure.

Before you begin

- An S3 object server is configured in an SVM containing NAS data.
- The NAS data conforms to the requirements for S3 client access.

About this task

You can configure S3 NAS buckets to specify any set of files and directories within the root directory of the SVM.

You can also set bucket policies that allow or disallow access to NAS data based on any combination of these parameters:

- Files and directories
- User and group permissions
- S3 operations

For example, you might want separate bucket policies that grant read-only data access to a large group of users, and another that allows a limited group to perform operations on a subset of that data.

Because S3 NAS “buckets” are mappings and not S3 buckets, the following properties of standard S3 buckets don’t apply to S3 NAS buckets.

- **aggr-list \ aggr-list-multiplier \ storage-service-level \ volume \ size \ exclude-aggr-list \ qos-policy-group**
  No volumes or qtree are created when configuring S3 NAS buckets.
- **role \ is -protected \ is -protected-on-ontap \ is -protected-on-cloud**
  S3 NAS buckets are not protected or mirrored using S3 SnapMirror, but will instead be using regular SnapMirror protection available at volume granularity.

- **versioning-state**
  NAS volumes usually have Snapshot technology available to save different versions. However, versioning is not currently available in S3 NAS buckets.
- **logical-used \ object-count**
  Equivalent statistics are available for NAS volumes through the volume commands.

System Manager

Add a new S3 NAS bucket on an NAS-enabled storage VM.

1. Click **Storage > Buckets**, then click **Add**.
2. Enter a name for the S3 NAS bucket and select the storage VM, do not enter a size, then click **More Options**.
3. Enter a valid path name or click Browse to select from a list of valid path names.
   When you enter a valid pathname, options that are not relevant to S3 NAS configuration are hidden.
4. If you have already mapped S3 users to NAS users and created groups, you can configure their permissions, then click **Save**.
   You must have already mapped S3 users to NAS users before configuring permissions in this step.

Otherwise, click **Save** to complete S3 NAS bucket configuration.

CLI

Create an S3 NAS bucket in an SVM containing NAS filesystems.
`vserver object-store-server bucket create -vserver` *`svm_name`* `-bucket` *`bucket_name`* `-type nas -nas-path` *`junction_path`* `[-comment` *`text`*`]`

Example:
```
cluster1::> vserver object-store-server bucket create -bucket testbucket -type nas -path /vol1
```

Enable S3 client users

To enable S3 client users to access NAS data, you must map S3 user names to corresponding NAS users, then grant them permission to access the NAS data using bucket service polices.

Before you begin

User names for client access – LINUX/UNIX, Windows and S3 client users – must already exist.

About this task

Mapping an S3 user name to a corresponding LINUX/UNIX or Windows user allows authorization checks on the NAS files to be honored when those files are accessed by S3 clients. S3 to NAS mappings are specified by providing an S3 user name *Pattern*, which can be expressed as a single name or a POSIX regular expression, and a LINUX/UNIX or Windows user name *Replacement*.

In case there is no name-mapping present, default name-mapping will be used, where the S3 user name itself will be used as the UNIX user name and Windows user name. You can modify the UNIX and Windows default user name mappings with the `vserver object-store-server modify` command.

Only local name-mapping configuration is supported; LDAP is not supported.

After S3 users are mapped to NAS users, you can grant permissions to users specifying the resources (directories and files) to which they have access and the actions they are allowed or not allowed to perform there.

System Manager

1. Create local name mappings for UNIX or Windows clients (or both).
   a. Click **Storage > Buckets**, then select the S3/NAS-enabled storage VM.
   b. Select **Settings**, then click → in **Name Mapping** (under **Host Users and Groups**).
   c. In the **S3 to Windows** or **S3 to UNIX** tiles (or both), click **Add**, then entered the desired **Pattern** (S3) and **Replacement** (NAS) user names.
2. Create a bucket policy to provide client access.
   a. Click **Storage > Buckets**, click ⋮ next to the desired S3 bucket, then click **Edit**.
   b. Click **Add** and supply the desired values.
      - **Principal** - Provide S3 user names or use the default (all users).
      - **Effect** - Select **Allow** or **Deny**.
      - **Actions** - Enter actions for these users and resources. The set of resource operations that the object store server currently supports for S3 NAS buckets are: GetObject, PutObject, DeleteObject, ListBucket, GetBucketAcl, GetObjectAcl, GetObjectTagging, PutObjectTagging, DeleteObjectTagging, GetBucketLocation, GetBucketVersioning, PutBucketVersioning and ListBucketVersions. Wildcards are accepted for this parameter.
      - **Resources** - Enter folder or file paths in which the actions are allowed or denied, or use the defaults (root directory of the bucket).

CLI

1. Create local name mappings for UNIX or Windows clients (or both).
   `vserver name-mapping create -vserver svm_name> -direction {s3-win|s3-unix} -position` *`integer`* `-pattern` *`s3_user_name`* `-replacement` *`nas_user_name`*
   - `-position` - priority number for mapping evaluation; enter 1 or 2.
   - `-pattern` - an S3 user name or a regular expression
   - `-replacement` - a windows or unix user name

   Examples
   ```
   vserver name-mapping create -direction s3-win -position 1 -pattern s3_user_1
   -replacement win_user_1 vserver name-mapping create -direction s3-unix
   -position 2 -pattern s3_user_1 -replacement unix_user_1
   ```

2. Create a bucket policy to provide client access.
   ```
   vserver object-store-server bucket policy add-statement -vserver svm_name
   -bucket bucket_name -effect {deny|allow} -action list_of_actions -principal
   list_of_users_or_groups -resource [-sid alphanumeric_text]
   ```
   - `-effect {deny|allow}` - specifies whether access is allowed or denied when a user requests an action.
   - `-action <Action>, …` - specifies resource operations that are allowed or denied. The set of resource operations that the object store server currently supports for S3 NAS buckets are: GetObject, PutObject, DeleteObject, ListBucket, GetBucketAcl, GetObjectAcl, GetObjectTagging, PutObjectTagging, DeleteObjectTagging, GetBucketLocation, GetBucketVersioning, PutBucketVersioning and ListBucketVersions. Wildcards are accepted for this parameter.
   - `-principal <Objectstore Principal>, …` - validates the user requesting access against

the object store server users or groups specified in this parameter.

- An object store server group is specified by adding a prefix group/ to the group name.
- `-principal` - (the hyphen character) grants access to all users.

- `-resource <text>, …` - specifies the bucket, folder, or object for which allow/deny permissions are set. Wildcards are accepted for this parameter.
- `[-sid <SID>]` - specifies an optional text comment for the object store server bucket policy statement.

Examples

```
cluster1::> vserver object-store-server bucket policy add-statement -bucket
testbucket -effect allow -action
GetObject,PutObject,DeleteObject,ListBucket,GetBucketAcl,GetObjectAcl,
GetBucketLocation,GetBucketPolicy,PutBucketPolicy,DeleteBucketPolicy
-principal user1 -resource testbucket,testbucket/* sid "FullAccessForUser1"

cluster1::> vserver object-store-server bucket policy statement create
-vserver vs1 -bucket bucket1 -effect allow -action GetObject -principal -
-resource bucket1/readme/* -sid "ReadAccessToReadmeForAllUsers"
```

Copyright information

Trademark information