Manage NFS with the CLI

ONTAP 9

NetApp
January 24, 2023

Table of Contents

Manage NFS with the CLI

NFS reference overview

ONTAP includes file access features available for the NFS protocol. You can enable an NFS server and export volumes or qtrees.

You perform these procedure under the following circumstances:

- You want to understand the range of ONTAP NFS protocol capabilities.
- You want to perform less common configuration and maintenance tasks, not basic NFS configuration.
- You want to use the command-line interface (CLI), not System Manager or an automated scripting tool.

Understand NAS file access

Namespaces and junction points

Namespaces and junction points overview

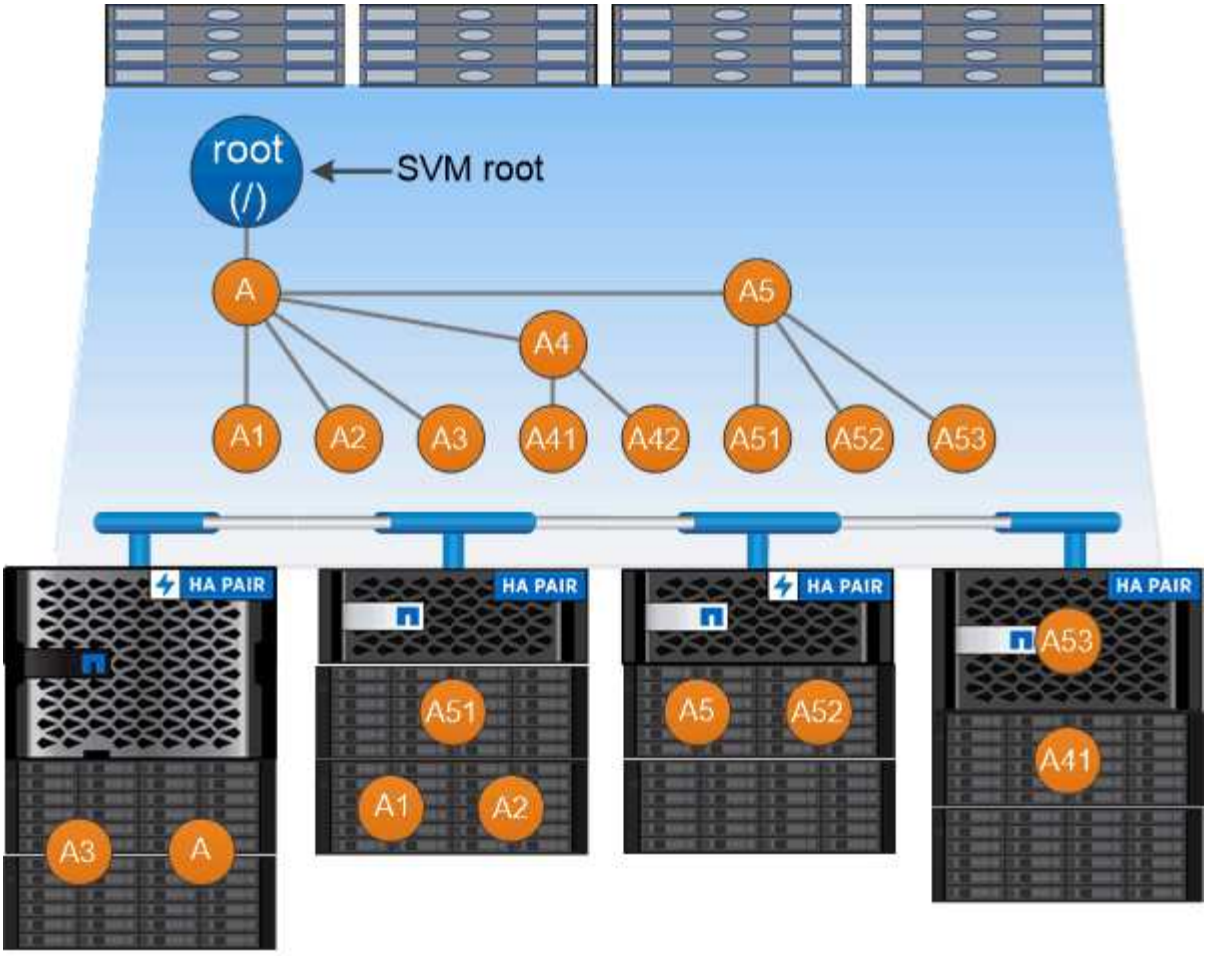
A NAS *namespace* is a logical grouping of volumes joined together at *junction points* to create a single file system hierarchy. A client with sufficient permissions can access files in the namespace without specifying the location of the files in storage. Junctioned volumes can reside anywhere in the cluster.

Rather than mounting every volume containing a file of interest, NAS clients mount an NFS *export* or access an SMB *share.* The export or share represents the entire namespace or an intermediate location within the namespace. The client accesses only the volumes mounted below its access point.

You can add volumes to the namespace as needed. You can create junction points directly below a parent volume junction or on a directory within a volume. A path to a volume junction for a volume named “vol3” might be `/vol1/vol2/vol3`, or `/vol1/dir2/vol3`, or even `/dir1/dir2/vol3`. The path is called the *junction path.*

Every SVM has a unique namespace. The SVM root volume is the entry point to the namespace hierarchy.

To ensure that data remains available in the event of a node outage or failover, you should create a *load-sharing mirror* copy for the SVM root volume.



A namespace is a logical grouping of volumes joined together at junction points to create a single file system hierarchy.

Example

The following example creates a volume named “home4” located on SVM vs1 that has a junction path `/eng/home`:

```
cluster1::> volume create -vserver vs1 -volume home4 -aggregate aggr1
-size 1g -junction-path /eng/home
[Job 1642] Job succeeded: Successful
```

What the typical NAS namespace architectures are

There are several typical NAS namespace architectures that you can use as you create your SVM name space. You can choose the namespace architecture that matches your business and workflow needs.

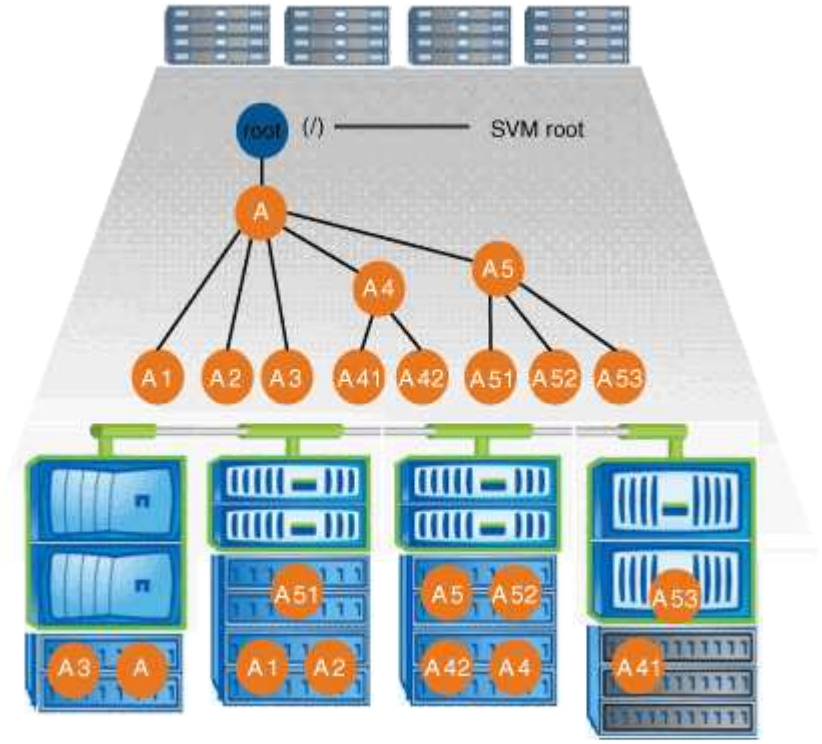
The top of the namespace is always the root volume, which is represented by a slash (/). The namespace architecture under the root falls into three basic categories:

- A single branched tree, with only a single junction to the root of the namespace

- Multiple branched trees, with multiple junction points to the root of the namespace
- Multiple stand-alone volumes, each with a separate junction point to the root of the name space

Namespace with single branched tree

An architecture with a single branched tree has a single insertion point to the root of the SVM namespace. The single insertion point can be either a junctioned volume or a directory beneath the root. All other volumes are mounted at junction points beneath the single insertion point (which can be a volume or a directory).

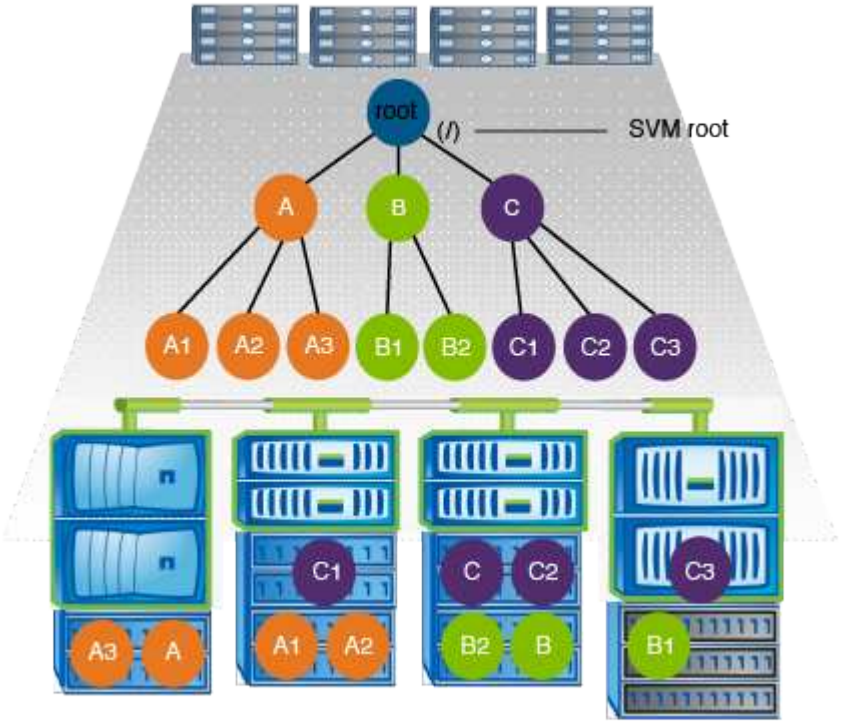


For example, a typical volume junction configuration with the above namespace architecture might look like the following configuration, where all volumes are junctioned below the single insertion point, which is a directory named “data”:

```
                    Junction                        Junction
Vserver Volume      Active   Junction Path          Path Source
------- ----------- -------- --------------------   -----------
vs1     corp1       true     /data/dir1/corp1       RW_volume
vs1     corp2       true     /data/dir1/corp2       RW_volume
vs1     data1       true     /data/data1            RW_volume
vs1     eng1        true     /data/data1/eng1       RW_volume
vs1     eng2        true     /data/data1/eng2       RW_volume
vs1     sales       true     /data/data1/sales      RW_volume
vs1     vol1        true     /data/vol1             RW_volume
vs1     vol2        true     /data/vol2             RW_volume
vs1     vol3        true     /data/vol3             RW_volume
vs1     vs1_root    -        /                      -
```

Namespace with multiple branched trees

An architecture with multiple branched trees has multiple insertion points to the root of the SVM namespace. The insertion points can be either junctioned volumes or directories beneath the root. All other volumes are mounted at junction points beneath the insertion points (which can be volumes or directories).



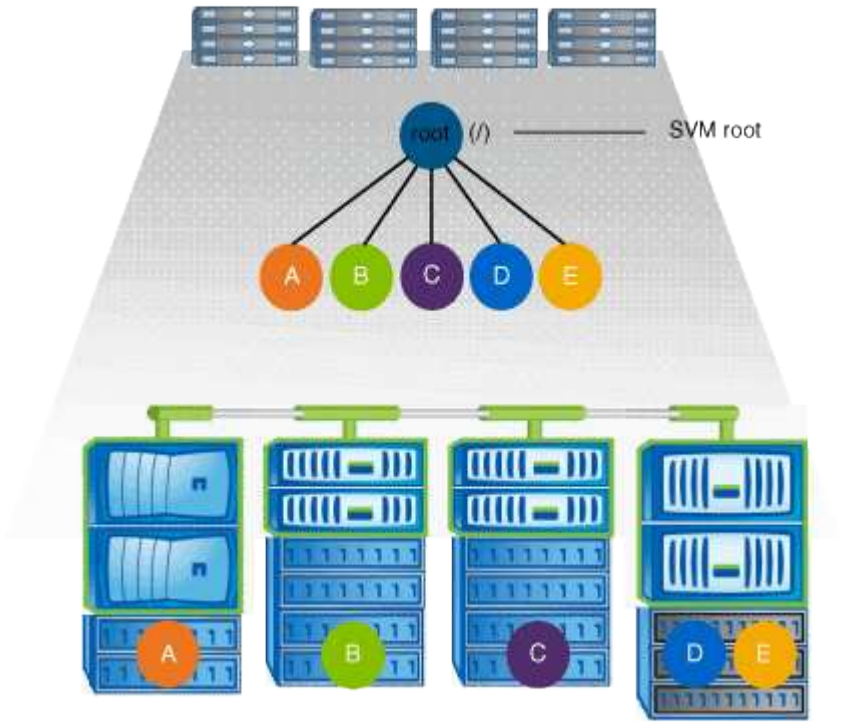
For example, a typical volume junction configuration with the above namespace architecture might look like the following configuration, where there are three insertion points to the root volume of the SVM. Two insertion points are directories named “data” and “projects”. One insertion point is a junctioned volume named “audit”:

```
                    Junction                        Junction
Vserver Volume      Active   Junction Path          Path Source
------- ----------- -------- --------------------   -----------
vs1     audit       true     /audit                 RW_volume
vs1     audit_logs1 true     /audit/logs1           RW_volume
vs1     audit_logs2 true     /audit/logs2           RW_volume
vs1     audit_logs3 true     /audit/logs3           RW_volume
vs1     eng         true     /data/eng              RW_volume
vs1     mktg1       true     /data/mktg1            RW_volume
vs1     mktg2       true     /data/mktg2            RW_volume
vs1     project1    true     /projects/project1     RW_volume
vs1     project2    true     /projects/project2     RW_volume
vs1     vs1_root    -        /                      -
```

Namespace with multiple stand-alone volumes

In an architecture with stand-alone volumes, every volume has an insertion point to the root of the SVM namespace; however, the volume is not junctioned below another volume. Each volume has a unique path,

and is either junctioned directly below the root or is junctioned under a directory below the root.



For example, a typical volume junction configuration with the above namespace architecture might look like the following configuration, where there are five insertion points to the root volume of the SVM, with each insertion point representing a path to one volume.

```
                    Junction                          Junction
Vserver Volume      Active   Junction Path            Path Source
------- ----------- -------- -------------------      -----------
vs1     eng         true     /eng                     RW_volume
vs1     mktg        true     /vol/mktg                RW_volume
vs1     project1    true     /project1                RW_volume
vs1     project2    true     /project2                RW_volume
vs1     sales       true     /sales                   RW_volume
vs1     vs1_root    -        /                        -
```

How ONTAP controls access to files

How ONTAP controls access to files overview

ONTAP controls access to files according to the authentication-based and file-based restrictions that you specify.

When a client connects to the storage system to access files, ONTAP has to perform two tasks:

- Authentication

  ONTAP has to authenticate the client by verifying the identity with a trusted source. In addition, the

authentication type of the client is one method that can be used to determine whether a client can access data when configuring export policies (optional for CIFS).

- Authorization

  ONTAP has to authorize the user by comparing the user's credentials with the permissions configured on the file or directory and determining what type of access, if any, to provide.

To properly manage file access control, ONTAP must communicate with external services such as NIS, LDAP, and Active Directory servers. Configuring a storage system for file access using CIFS or NFS requires setting up the appropriate services depending on your environment in ONTAP.

Authentication-based restrictions

With authentication-based restrictions, you can specify which client machines and which users can connect to the storage virtual machine (SVM).

ONTAP supports Kerberos authentication from both UNIX and Windows servers.

File-based restrictions

ONTAP evaluates three levels of security to determine whether an entity is authorized to perform a requested action on files and directories residing on an SVM. Access is determined by the effective permissions after evaluation of the three security levels.

Any storage object can contain up to three types of security layers:

- Export (NFS) and share (SMB) security

  Export and share security applies to client access to a given NFS export or SMB share. Users with administrative privileges can manage export and share-level security from SMB and NFS clients.

- Storage-Level Access Guard file and directory security

  Storage-Level Access Guard security applies to SMB and NFS client access to SVM volumes. Only NTFS access permissions are supported. For ONTAP to perform security checks on UNIX users for access to data on volumes for which Storage-Level Access Guard has been applied, the UNIX user must map to a Windows user on the SVM that owns the volume.

  > If you view the security settings on a file or directory from an NFS or SMB client, you will not see Storage-Level Access Guard security. Storage-Level Access Guard security cannot be revoked from a client, even by a system (Windows or UNIX) administrator.

- NTFS, UNIX, and NFSv4 native file-level security

  Native file-level security exists on the file or directory that represents the storage object. You can set file-level security from a client. File permissions are effective regardless of whether SMB or NFS is used to access the data.

How ONTAP handles NFS client authentication

How ONTAP handles NFS client authentication overview

NFS clients must be properly authenticated before they can access data on the SVM. ONTAP authenticates the clients by checking their UNIX credentials against the name services that you configure.

When an NFS client connects to the SVM, ONTAP obtains the UNIX credentials for the user by checking different name services, depending on the name services configuration of the SVM. ONTAP can check credentials for local UNIX accounts, NIS domains, and LDAP domains. At least one of them must be configured so that ONTAP can successfully authenticate the user. You can specify multiple name services and the order in which ONTAP searches them.

In a pure NFS environment with UNIX volume security styles, this configuration is sufficient to authenticate and provide the proper file access for a user connecting from an NFS client.

If you are using mixed, NTFS, or unified volume security styles, ONTAP must obtain a SMB user name for the UNIX user for authentication with a Windows domain controller. This can happen either by mapping individual users using local UNIX accounts or LDAP domains, or by using a default SMB user instead. You can specify which name services ONTAP searches in which order, or specify a default SMB user.

How ONTAP uses name services

ONTAP uses name services to obtain information about users and clients. ONTAP uses this information to authenticate users accessing data on or administering the storage system, and to map user credentials in a mixed environment.

When you configure the storage system, you must specify what name services you want ONTAP to use for obtaining user credentials for authentication. ONTAP supports the following name services:

- Local users (file)
- External NIS domains (NIS)
- External LDAP domains (LDAP)

You use the `vserver services name-service ns-switch` command family to configure SVMs with the sources to search for network information and the order in which to search them. These commands provide the equivalent functionality of the `/etc/nsswitch.conf` file on UNIX systems.

When an NFS client connects to the SVM, ONTAP checks the specified name services to obtain the UNIX credentials for the user. If name services are configured correctly and ONTAP can obtain the UNIX credentials, ONTAP successfully authenticates the user.

In an environment with mixed security styles, ONTAP might have to map user credentials. You must configure name services appropriately for your environment to allow ONTAP to properly map user credentials.

ONTAP also uses name services for authenticating SVM administrator accounts. You must keep this in mind when configuring or modifying the name service switch to avoid accidentally disabling authentication for SVM administrator accounts. For more information about SVM administration users, see xref:./nfs-admin/../authentication/index.html[Administrator authentication and RBAC.

How ONTAP grants SMB file access from NFS clients

ONTAP uses Windows NT File System (NTFS) security semantics to determine whether

a UNIX user, on an NFS client, has access to a file with NTFS permissions.

ONTAP does this by converting the user's UNIX User ID (UID) into a SMB credential, and then using the SMB credential to verify that the user has access rights to the file. A SMB credential consists of a primary Security Identifier (SID), usually the user's Windows user name, and one or more group SIDs that correspond to Windows groups of which the user is a member.

The time ONTAP takes converting the UNIX UID into a SMB credential can be from tens of milliseconds to hundreds of milliseconds because the process involves contacting a domain controller. ONTAP maps the UID to the SMB credential and enters the mapping in a credential cache to reduce the verification time caused by the conversion.

How the NFS credential cache works

When an NFS user requests access to NFS exports on the storage system, ONTAP must retrieve the user credentials either from external name servers or from local files to authenticate the user. ONTAP then stores these credentials in an internal credential cache for later reference. Understanding how the NFS credential caches works enables you to handle potential performance and access issues.

Without the credential cache, ONTAP would have to query name services every time an NFS user requested access. On a busy storage system that is accessed by many users, this can quickly lead to serious performance problems, causing unwanted delays or even denials to NFS client access.

With the credential cache, ONTAP retrieves the user credentials and then stores them for a predetermined amount of time for quick and easy access should the NFS client send another request. This method offers the following advantages:

- It eases the load on the storage system by handling fewer requests to external name servers (such as NIS or LDAP).
- It eases the load on external name servers by sending fewer requests to them.
- It speeds up user access by eliminating the wait time for obtaining credentials from external sources before the user can be authenticated.

ONTAP stores both positive and negative credentials in the credential cache. Positive credentials means that the user was authenticated and granted access. Negative credentials means that the user was not authenticated and was denied access.

By default, ONTAP stores positive credentials for 24 hours; that is, after initially authenticating a user, ONTAP uses the cached credentials for any access requests by that user for 24 hours. If the user requests access after 24 hours, the cycle starts over: ONTAP discards the cached credentials and obtains the credentials again from the appropriate name service source. If the credentials changed on the name server during the previous 24 hours, ONTAP caches the updated credentials for use for the next 24 hours.

By default, ONTAP stores negative credentials for two hours; that is, after initially denying access to a user, ONTAP continues to deny any access requests by that user for two hours. If the user requests access after 2 hours, the cycle starts over: ONTAP obtains the credentials again from the appropriate name service source. If the credentials changed on the name server during the previous two hours, ONTAP caches the updated credentials for use for the next two hours.

Create and manage data volumes in NAS namespaces

Create data volumes with specified junction points

You can specify the junction point when you create a data volume. The resultant volume is automatically mounted at the junction point and is immediately available to configure for NAS access.

What you'll need

The aggregate in which you want to create the volume must already exist.

> The following characters cannot be used in the junction path: * # " > < | ? \
>
> In addition, the junction path length cannot be more than 255 characters.

Steps

1. Create the volume with a junction point:

   ```
   volume create -vserver vserver_name -volume volume_name -aggregate
   aggregate_name -size {integer[KB|MB|GB|TB|PB]} -security-style
   {ntfs|unix|mixed} -junction-path junction_path
   ```

   The junction path must start with the root (/) and can contain both directories and junctioned volumes. The junction path does not need to contain the name of the volume. Junction paths are independent of the volume name.

   Specifying a volume security style is optional. If you do not specify a security style, ONTAP creates the volume with the same security style that is applied to the root volume of the storage virtual machine (SVM). However, the root volume's security style might not be the security style you want applied to the data volume you create. The recommendation is to specify the security style when you create the volume to minimize difficult-to-troubleshoot file-access issues.

   The junction path is case insensitive; `/ENG` is the same as `/eng`. If you create a CIFS share, Windows treats the junction path as if it is case sensitive. For example, if the junction is `/ENG`, the path of a SMB share must start with `/ENG`, not `/eng`.

   There are many optional parameters that you can use to customize a data volume. To learn more about them, see the man pages for the `volume create` command.

2. Verify that the volume was created with the desired junction point:

   ```
   volume show -vserver vserver_name -volume volume_name -junction
   ```

Example

The following example creates a volume named "home4" located on SVM vs1 that has a junction path `/eng/home`:

```
cluster1::> volume create -vserver vs1 -volume home4 -aggregate aggr1
-size 1g -junction-path /eng/home
[Job 1642] Job succeeded: Successful

cluster1::> volume show -vserver vs1 -volume home4 -junction
                  Junction                 Junction
Vserver   Volume  Active   Junction Path   Path Source
--------- ------- -------- --------------- -----------
vs1       home4   true     /eng/home       RW_volume
```

Create data volumes without specifying junction points

You can create a data volume without specifying a junction point. The resultant volume is not automatically mounted, and is not available to configure for NAS access. You must mount the volume before you can configure SMB shares or NFS exports for that volume.

What you'll need

The aggregate in which you want to create the volume must already exist.

Steps

1. Create the volume without a junction point by using the following command:

   ```
   volume create -vserver vserver_name -volume volume_name -aggregate
   aggregate_name -size {integer[KB|MB|GB|TB|PB]} -security-style
   {ntfs|unix|mixed}
   ```

   Specifying a volume security style is optional. If you do not specify a security style, ONTAP creates the volume with the same security style that is applied to the root volume of the storage virtual machine (SVM). However, the root volume's security style might not be the security style you want applied to the data volume. The recommendation is to specify the security style when you create the volume to minimize difficult-to-troubleshoot file-access issues.

   There are many optional parameters that you can use to customize a data volume. To learn more about them, see the man pages for the `volume create` command.

2. Verify that the volume was created without a junction point:

   ```
   volume show -vserver vserver_name -volume volume_name -junction
   ```

Example

The following example creates a volume named "sales" located on SVM vs1 that is not mounted at a junction point:

```
cluster1::> volume create -vserver vs1 -volume sales -aggregate aggr3
-size 20GB
[Job 3406] Job succeeded: Successful

cluster1::> volume show -vserver vs1 -junction
                     Junction                 Junction
Vserver   Volume     Active   Junction Path   Path Source
--------- ---------- -------- --------------- -----------
vs1       data       true     /data           RW_volume
vs1       home4      true     /eng/home       RW_volume
vs1       vs1_root   -        /               -
vs1       sales      -        -               -
```

Mount or unmount existing volumes in the NAS namespace

A volume must be mounted on the NAS namespace before you can configure NAS client access to data contained in the storage virtual machine (SVM) volumes. You can mount a volume to a junction point if it is not currently mounted. You can also unmount volumes.

About this task

If you unmount and offline a volume, all data within the junction point, including data in volumes with junction points contained within the unmounted volume's namespace, are inaccessible to NAS clients.

> To discontinue NAS client access to a volume, it is not sufficient to simply unmount the volume. You must offline the volume, or take other steps to ensure that client-side file handle caches are invalidated. For more information, see the following Knowledge Base article:
>
> NFSv3 clients still have access to a volume after being removed from the namespace in ONTAP

When you unmount and offline a volume, data within the volume is not lost. Additionally, existing volume export policies and SMB shares created on the volume or on directories and junction points within the unmounted volume are retained. If you remount the unmounted volume, NAS clients can access the data contained within the volume using existing export policies and SMB shares.

Steps

1. Perform the desired action:

If you want to…	Enter the commands…
Mount a volume	`volume mount -vserver svm_name -volume volume_name -junction-path junction_path`
Unmount a volume	`volume unmount -vserver svm_name -volume volume_name volume offline -vserver svm_name -volume volume_name`

2. Verify that the volume is in the desired mount state:

```
volume show -vserver vserver_name -volume volume_name -fields state,junction-
path,junction-active
```

Examples

The following example mounts a volume named “sales” located on SVM vs1 to the junction point `/sales`:

```
cluster1::> volume mount -vserver vs1 -volume sales -junction-path /sales

cluster1::> volume show -vserver vs1 state,junction-path,junction-active

vserver   volume     state     junction-path   junction-active
--------- ---------- -------- --------------- ----------------
vs1       data       online    /data           true
vs1       home4      online    /eng/home       true
vs1       sales      online    /sales          true
```

The following example unmounts and offlines a volume named “data” located on SVM vs1:

```
cluster1::> volume unmount -vserver vs1 -volume data
cluster1::> volume offline -vserver vs1 -volume data

cluster1::> volume show -vserver vs1 -fields state,junction-path,junction-
active

vserver   volume     state     junction-path   junction-active
--------- ---------- --------- --------------- ---------------
vs1       data       offline   -               -
vs1       home4      online    /eng/home       true
vs1       sales      online    /sales          true
```

Display volume mount and junction point information

You can display information about mounted volumes for storage virtual machines (SVMs) and the junction points to which the volumes are mounted. You can also determine which volumes are not mounted to a junction point. You can use this information to understand and manage your SVM namespace.

Step

1. Perform the desired action:

If you want to display…	Enter the command…

Summary information about mounted and unmounted volumes on the SVM	`volume show -vserver vserver_name -junction`
Detailed information about mounted and unmounted volumes on the SVM	`volume show -vserver vserver_name -volume volume_name -instance`
Specific information about mounted and unmounted volumes on the SVM	a. If necessary, you can display valid fields for the `-fields` parameter by using the following command: `volume show -fields ?` b. Display the desired information by using the `-fields` parameter: `volume show -vserver vserver_name -fields fieldname,…`

Examples

The following example displays a summary of mounted and unmounted volumes on SVM vs1:

```
cluster1::> volume show -vserver vs1 -junction
                    Junction                 Junction
Vserver   Volume    Active   Junction Path   Path Source
--------- --------- -------- --------------- -----------
vs1       data      true     /data           RW_volume
vs1       home4     true     /eng/home       RW_volume
vs1       vs1_root  -        /               -
vs1       sales     true     /sales          RW_volume
```

The following example displays information about specified fields for volumes located on SVM vs2:

```
cluster1::> volume show -vserver vs2 -fields
vserver,volume,aggregate,size,state,type,security-style,junction-
path,junction-parent,node
vserver volume   aggregate size state  type security-style junction-path
junction-parent node
------- ------   --------- ---- ------ ---- -------------- -------------
--------------- -----
vs2     data1    aggr3     2GB  online RW   unix           -             -
node3
vs2     data2    aggr3     1GB  online RW   ntfs           /data2
vs2_root        node3
vs2     data2_1  aggr3     8GB  online RW   ntfs           /data2/d2_1
data2           node3
vs2     data2_2  aggr3     8GB  online RW   ntfs           /data2/d2_2
data2           node3
vs2     pubs     aggr1     1GB  online RW   unix           /publications
vs2_root        node1
vs2     images   aggr3     2TB  online RW   ntfs           /images
vs2_root        node3
vs2     logs     aggr1     1GB  online RW   unix           /logs
vs2_root        node1
vs2     vs2_root aggr3     1GB  online RW   ntfs           /             -
node3
```

Configure security styles

How security styles affect data access

What the security styles and their effects are

There are four different security styles: UNIX, NTFS, mixed, and unified. Each security style has a different effect on how permissions are handled for data. You must understand the different effects to ensure that you select the appropriate security style for your purposes.

It is important to understand that security styles do not determine what client types can or cannot access data. Security styles only determine the type of permissions ONTAP uses to control data access and what client type can modify these permissions.

For example, if a volume uses UNIX security style, SMB clients can still access data (provided that they properly authenticate and authorize) due to the multiprotocol nature of ONTAP. However, ONTAP uses UNIX permissions that only UNIX clients can modify using native tools.

Security style	Clients that can modify permissions	Permissions that clients can use	Resulting effective security style	Clients that can access files
Unix	NFS	NFSv3 mode bits	Unix	NFS and SMB
		NFSv4.x ACLs		
NTFS	SMB	NTFS ACLs	NTFS	
Mixed	NFS or SMB	NFSv3 mode bits	UNIX	
		NFSv4.ACLs		
		NTFS ACLs	NTFS	
Unified (For infinite volumes only, in ONTAP 9.4 and earlier releases.)	NFS or SMB	NFSv3 mode bits	Unix	
		NFSv4.1 ACLs		
		NTFS ACLs	NTFS	

FlexVol volumes support UNIX, NTFS, and mixed security styles. When the security style is mixed or unified, the effective permissions depend on the client type that last modified the permissions because users set the security style on an individual basis. If the last client that modified permissions was an NFSv3 client, the permissions are UNIX NFSv3 mode bits. If the last client was an NFSv4 client, the permissions are NFSv4 ACLs. If the last client was an SMB client, the permissions are Windows NTFS ACLs.

The unified security style is only available with infinite volumes, which are no longer supported in ONTAP 9.5 and later releases. For more information, see FlexGroup volumes management overview.

Beginning with ONTAP 9.2, the `show-effective-permissions` parameter to the `vserver security file-directory` command enables you to display effective permissions granted to a Windows or UNIX user on the specified file or folder path. In addition, the optional parameter `-share-name` enables you to display the effective share permission.

ONTAP initially sets some default file permissions. By default, the effective security style on all data in UNIX, mixed, and unified security style volumes is UNIX and the effective permissions type is UNIX mode bits (0755 unless specified otherwise) until configured by a client as allowed by the default security style. By default, the effective security style on all data in NTFS security style volumes is NTFS and has an ACL allowing full control to everyone.

Where and when to set security styles

Security styles can be set on FlexVol volumes (both root or data volumes) and qtrees. Security styles can be set manually at the time of creation, inherited automatically, or changed at a later time.

Decide which security style to use on SVMs

To help you decide which security style to use on a volume, you should consider two factors. The primary factor is the type of administrator that manages the file system. The secondary factor is the type of user or service that accesses the data on the volume.

When you configure the security style on a volume, you should consider the needs of your environment to ensure that you select the best security style and avoid issues with managing permissions. The following

considerations can help you decide:

Security style	Choose if…
UNIX	• The file system is managed by a UNIX administrator. • The majority of users are NFS clients. • An application accessing the data uses a UNIX user as the service account.
NTFS	• The file system is managed by a Windows administrator. • The majority of users are SMB clients. • An application accessing the data uses a Windows user as the service account.
Mixed	• The file system is managed by both UNIX and Windows administrators and users consist of both NFS and SMB clients.

How security style inheritance works

If you do not specify the security style when creating a new FlexVol volume or a qtree, it inherits its security style in different ways.

Security styles are inherited in the following manner:

- A FlexVol volume inherits the security style of the root volume of its containing SVM.
- A qtree inherits the security style of its containing FlexVol volume.
- A file or directory inherits the security style of its containing FlexVol volume or qtree.

How ONTAP preserves UNIX permissions

When files in a FlexVol volume that currently have UNIX permissions are edited and saved by Windows applications, ONTAP can preserve the UNIX permissions.

When applications on Windows clients edit and save files, they read the security properties of the file, create a new temporary file, apply those properties to the temporary file, and then give the temporary file the original file name.

When Windows clients perform a query for the security properties, they receive a constructed ACL that exactly represents the UNIX permissions. The sole purpose of this constructed ACL is to preserve the file's UNIX permissions as files are updated by Windows applications to ensure that the resulting files have the same UNIX permissions. ONTAP does not set any NTFS ACLs using the constructed ACL.

Manage UNIX permissions using the Windows Security tab

If you want to manipulate UNIX permissions of files or folders in mixed security-style volumes or qtrees on SVMs, you can use the Security tab on Windows clients. Alternatively, you can use applications that can query and set Windows ACLs.

- Modifying UNIX permissions

  You can use the Windows Security tab to view and change UNIX permissions for a mixed security-style

volume or qtree. If you use the main Windows Security tab to change UNIX permissions, you must first remove the existing ACE you want to edit (this sets the mode bits to 0) before you make your changes. Alternatively, you can use the Advanced editor to change permissions.

If mode permissions are used, you can directly change the mode permissions for the listed UID, GID, and others (everyone else with an account on the computer). For example, if the displayed UID has r-x permissions, you can change the UID permissions to rwx.

- Changing UNIX permissions to NTFS permissions

  You can use the Windows Security tab to replace UNIX security objects with Windows security objects on a mixed security-style volume or qtree where the files and folders have a UNIX effective security style.

  You must first remove all listed UNIX permission entries before you can replace them with the desired Windows User and Group objects. You can then configure NTFS-based ACLs on the Windows User and Group objects. By removing all UNIX security objects and adding only Windows Users and Groups to a file or folder in a mixed security-style volume or qtree, you change the effective security style on the file or folder from UNIX to NTFS.

  When changing permissions on a folder, the default Windows behavior is to propagate these changes to all subfolders and files. Therefore, you must change the propagation choice to the desired setting if you do not want to propagate a change in security style to all child folders, subfolders, and files.

Configure security styles on SVM root volumes

You configure the storage virtual machine (SVM) root volume security style to determine the type of permissions used for data on the root volume of the SVM.

Steps

1. Use the `vserver create` command with the `-rootvolume-security-style` parameter to define the security style.

   The possible options for the root volume security style are `unix`, `ntfs`, or `mixed`.

2. Display and verify the configuration, including the root volume security style of the SVM you created:

   `vserver show -vserver` *`vserver_name`*

Configure security styles on FlexVol volumes

You configure the FlexVol volume security style to determine the type of permissions used for data on FlexVol volumes of the storage virtual machine (SVM).

Steps

1. Perform one of the following actions:

If the FlexVol volume…	Use the command…
Does not yet exist	`volume create` and include the `-security-style` parameter to specify the security style.

Already exists	`volume modify` and include the `-security-style` parameter to specify the security style.

The possible options for the FlexVol volume security style are `unix`, `ntfs`, or `mixed`.

If you do not specify a security style when creating a FlexVol volume, the volume inherits the security style of the root volume.

For more information about the `volume create` or `volume modify` commands, see Logical storage management.

2. To display the configuration, including the security style of the FlexVol volume you created, enter the following command:

```
volume show -volume volume_name -instance
```

Configure security styles on qtrees

You configure the qtree volume security style to determine the type of permissions used for data on qtrees.

Steps

1. Perform one of the following actions:

If the qtree…	Use the command…
Does not exist yet	`volume qtree create` and include the `-security-style` parameter to specify the security style.
Already exists	`volume qtree modify` and include the `-security-style` parameter to specify the security style.

The possible options for the qtree security style are `unix`, `ntfs`, or `mixed`.

If you do not specify a security style when creating a qtree, the default security style is `mixed`.

For more information about the `volume qtree create` or `volume qtree modify` commands, see Logical storage management.

2. To display the configuration, including the security style of the qtree you created, enter the following command: `volume qtree show -qtree` *`qtree_name`* `-instance`

Set up file access using NFS

Set up file access using NFS overview

You must complete a number of steps to allow clients access to files on storage virtual machines (SVMs) using NFS. There are some additional steps that are optional

depending on the current configuration of your environment.

For clients to be able to access files on SVMs using NFS, you must complete the following tasks:

1. Enable the NFS protocol on the SVM.

   You must configure the SVM to allow data access from clients over NFS.

2. Create an NFS server on the SVM.

   An NFS server is a logical entity on the SVM that enables the SVM to serve files over NFS. You must create the NFS server and specify the NFS protocol versions you want to allow.

3. Configure export policies on the SVM.

   You must configure export policies to make volumes and qtrees available to clients.

4. Configure the NFS server with the appropriate security and other settings depending on the network and storage environment.

   This step might include configuring Kerberos, LDAP, NIS, name mappings, and local users.

Secure NFS access using export policies

How export policies control client access to volumes or qtrees

Export policies contain one or more *export rules* that process each client access request. The result of the process determines whether the client is denied or granted access and what level of access. An export policy with export rules must exist on the storage virtual machine (SVM) for clients to access data.

You associate exactly one export policy with each volume or qtree to configure client access to the volume or qtree. The SVM can contain multiple export policies. This enables you to do the following for SVMs with multiple volumes or qtrees:

- Assign different export policies to each volume or qtree of the SVM for individual client access control to each volume or qtree in the SVM.
- Assign the same export policy to multiple volumes or qtrees of the SVM for identical client access control without having to create a new export policy for each volume or qtree.

If a client makes an access request that is not permitted by the applicable export policy, the request fails with a permission-denied message. If a client does not match any rule in the export policy, then access is denied. If an export policy is empty, then all accesses are implicitly denied.

You can modify an export policy dynamically on a system running ONTAP.

Default export policy for SVMs

Each SVM has a default export policy that contains no rules. An export policy with rules must exist before clients can access data on the SVM. Each FlexVol volume contained in the SVM must be associated with an export policy.

When you create an SVM, the storage system automatically creates a default export policy called `default` for the root volume of the SVM. You must create one or more rules for the default export policy before clients can access data on the SVM. Alternatively, you can create a custom export policy with rules. You can modify and rename the default export policy, but you cannot delete the default export policy.

When you create a FlexVol volume in its containing SVM, the storage system creates the volume and associates the volume with the default export policy for the root volume of the SVM. By default, each volume created in the SVM is associated with the default export policy for the root volume. You can use the default export policy for all volumes contained in the SVM, or you can create a unique export policy for each volume. You can associate multiple volumes with the same export policy.

How export rules work

Export rules are the functional elements of an export policy. Export rules match client access requests to a volume against specific parameters you configure to determine how to handle the client access requests.

An export policy must contain at least one export rule to allow access to clients. If an export policy contains more than one rule, the rules are processed in the order in which they appear in the export policy. The rule order is dictated by the rule index number. If a rule matches a client, the permissions of that rule are used and no further rules are processed. If no rules match, the client is denied access.

You can configure export rules to determine client access permissions using the following criteria:

- The file access protocol used by the client sending the request, for example, NFSv4 or SMB.
- A client identifier, for example, host name or IP address.

  The maximum size for the `-clientmatch` field is 4096 characters.

- The security type used by the client to authenticate, for example, Kerberos v5, NTLM, or AUTH_SYS.

If a rule specifies multiple criteria, the client must match all of them for the rule to apply.

> Beginning with ONTAP 9.3, you can enable export policy configuration checking as a background job that records any rules violations in an error rule list. The `vserver export-policy config-checker` commands invoke the checker and display results, which you can use to verify your configuration and delete erroneous rules from the policy.
>
> The commands only validate export configuration for host names, netgroups, and anonymous users.

Example

The export policy contains an export rule with the following parameters:

- `-protocol nfs3`
- `-clientmatch 10.1.16.0/255.255.255.0`
- `-rorule any`
- `-rwrule any`

The client access request is sent using the NFSv3 protocol and the client has the IP address 10.1.17.37.

Even though the client access protocol matches, the IP address of the client is in a different subnet from the one specified in the export rule. Therefore, client matching fails and this rule does not apply to this client.

Example

The export policy contains an export rule with the following parameters:

- `-protocol nfs`
- `-clientmatch 10.1.16.0/255.255.255.0`
- `-rorule any`
- `-rwrule any`

The client access request is sent using the NFSv4 protocol and the client has the IP address 10.1.16.54.

The client access protocol matches and the IP address of the client is in the specified subnet. Therefore, client matching is successful and this rule applies to this client. The client gets read-write access regardless of its security type.

Example

The export policy contains an export rule with the following parameters:

- `-protocol nfs3`
- `-clientmatch 10.1.16.0/255.255.255.0`
- `-rorule any`
- `-rwrule krb5,ntlm`

Client #1 has the IP address 10.1.16.207, sends an access request using the NFSv3 protocol, and authenticated with Kerberos v5.

Client #2 has the IP address 10.1.16.211, sends an access request using the NFSv3 protocol, and authenticated with AUTH_SYS.

The client access protocol and IP address matches for both clients. The read-only parameter allows read-only access to all clients regardless of the security type they authenticated with. Therefore both clients get read-only access. However, only client #1 gets read-write access because it used the approved security type Kerberos v5 to authenticate. Client #2 does not get read-write access.

Manage clients with an unlisted security type

When a client presents itself with a security type that is not listed in an access parameter of an export rule, you have the choice of either denying access to the client or mapping it to the anonymous user ID instead by using the option `none` in the access parameter.

A client might present itself with a security type that is not listed in an access parameter because it was authenticated with a different security type or was not authenticated at all (security type AUTH_NONE). By default, the client is automatically denied access to that level. However, you can add the option `none` to the access parameter. As a result, clients with an unlisted security style are mapped to the anonymous user ID instead. The `-anon` parameter determines what user ID is assigned to those clients. The user ID specified for the `-anon` parameter must be a valid user that is configured with permissions you deem appropriate for the anonymous user.

Valid values for the `-anon` parameter range from `0` to `65535`.

User ID assigned to `-anon`	**Resulting handling of client access requests**
`0 - 65533`	The client access request is mapped to the anonymous user ID and gets access depending on the permissions configured for this user.
`65534`	The client access request is mapped to the user nobody and gets access depending on the permissions configured for this user. This is the default.
`65535`	The access request from any client is denied when mapped to this ID and the client presents itself with security type AUTH_NONE. The access request from clients with user ID 0 is denied when mapped to this ID and the client presents itself with any other security type.

When using the option `none`, it is important to remember that the read-only parameter is processed first. Consider the following guidelines when configuring export rules for clients with unlisted security types:

Read-only includes `none`	**Read-write includes** `none`	**Resulting access for clients with unlisted security types**
No	No	Denied
No	Yes	Denied because read-only is processed first
Yes	No	Read-only as anonymous
Yes	Yes	Read-write as anonymous

Example

The export policy contains an export rule with the following parameters:

- `-protocol nfs3`
- `-clientmatch 10.1.16.0/255.255.255.0`
- `-rorule sys,none`
- `-rwrule any`
- `-anon 70`

Client #1 has the IP address 10.1.16.207, sends an access request using the NFSv3 protocol, and authenticated with Kerberos v5.

Client #2 has the IP address 10.1.16.211, sends an access request using the NFSv3 protocol, and

authenticated with AUTH_SYS.

Client #3 has the IP address 10.1.16.234, sends an access request using the NFSv3 protocol, and did not authenticate (meaning security type AUTH_NONE).

The client access protocol and IP address matches for all three clients. The read-only parameter allows read-only access to clients with their own user ID that authenticated with AUTH_SYS. The read-only parameter allows read-only access as the anonymous user with user ID 70 to clients that authenticated using any other security type. The read-write parameter allows read-write access to any security type, but in this case only applies to clients already filtered by the read-only rule.

Therefore, clients #1 and #3 get read-write access only as the anonymous user with user ID 70. Client #2 gets read-write access with its own user ID.

Example

The export policy contains an export rule with the following parameters:

- `-protocol nfs3`
- `-clientmatch 10.1.16.0/255.255.255.0`
- `-rorule sys,none`
- `-rwrule none`
- `-anon 70`

Client #1 has the IP address 10.1.16.207, sends an access request using the NFSv3 protocol, and authenticated with Kerberos v5.

Client #2 has the IP address 10.1.16.211, sends an access request using the NFSv3 protocol, and authenticated with AUTH_SYS.

Client #3 has the IP address 10.1.16.234, sends an access request using the NFSv3 protocol, and did not authenticate (meaning security type AUTH_NONE).

The client access protocol and IP address matches for all three clients. The read-only parameter allows read-only access to clients with their own user ID that authenticated with AUTH_SYS. The read-only parameter allows read-only access as the anonymous user with user ID 70 to clients that authenticated using any other security type. The read-write parameter allows read-write access only as the anonymous user.

Therefore, client #1 and client #3 get read-write access only as the anonymous user with user ID 70. Client #2 gets read-only access with its own user ID but is denied read-write access.

How security types determine client access levels

The security type that the client authenticated with plays a special role in export rules. You must understand how the security type determines the levels of access the client gets to a volume or qtree.

The three possible access levels are as follows:

1. Read-only
2. Read-write
3. Superuser (for clients with user ID 0)

Because the access level by security type is evaluated in this order, you must observe the following rules when constructing access level parameters in export rules:

For a client to get access level…	These access parameters must match the client's security type…
Normal user read-only	Read-only (`-rorule`)
Normal user read-write	Read-only (`-rorule`) and read-write (`-rwrule`)
Superuser read-only	Read-only (`-rorule`) and `-superuser`
Superuser read-write	Read-only (`-rorule`) and read-write (`-rwrule`) and `-superuser`

The following are valid security types for each of these three access parameters:

- `any`
- `none`
- `never`

  This security type is not valid for use with the `-superuser` parameter.

- `krb5`
- `krb5i`
- `krb5p`
- `ntlm`
- `sys`

When matching a client's security type against each of the three access parameters, there are three possible outcomes:

If the client's security type…	Then the client…
Matches the one specified in the access parameter.	Gets access for that level with its own user ID.
Does not match the one specified, but the access parameter includes the option `none`.	Gets access for that level but as the anonymous user with the user ID specified by the `-anon` parameter.
Does not match the one specified and the access parameter does not include the option `none`.	Does not get any access for that level.This does not apply to the `-superuser` parameter because it always includes `none` even when not specified.

Example

The export policy contains an export rule with the following parameters:

- `-protocol nfs3`
- `-clientmatch 10.1.16.0/255.255.255.0`
- `-rorule any`
- `-rwrule sys,krb5`
- `-superuser krb5`

Client #1 has the IP address 10.1.16.207, has user ID 0, sends an access request using the NFSv3 protocol, and authenticated with Kerberos v5.

Client #2 has the IP address 10.1.16.211, has user ID 0, sends an access request using the NFSv3 protocol, and authenticated with AUTH_SYS.

Client #3 has the IP address 10.1.16.234, has user ID 0, sends an access request using the NFSv3 protocol, and did not authenticate (AUTH_NONE).

The client access protocol and IP address matches all three clients. The read-only parameter allows read-only access to all clients regardless of security type. The read-write parameter allows read-write access to clients with their own user ID that authenticated with AUTH_SYS or Kerberos v5. The superuser parameter allows superuser access to clients with user ID 0 that authenticated with Kerberos v5.

Therefore, client #1 gets superuser read-write access because it matches all three access parameters. Client #2 gets read-write access but not superuser access. Client #3 gets read-only access but not superuser access.

Manage superuser access requests

When you configure export policies, you need to consider what you want to happen if the storage system receives a client access request with user ID 0, meaning as a superuser, and set up your export rules accordingly.

In the UNIX world, a user with the user ID 0 is known as the superuser, typically called root, who has unlimited access rights on a system. Using superuser privileges can be dangerous for several reasons, including breach of system and data security.

By default, ONTAP maps clients presenting with user ID 0 to the anonymous user. However, you can specify the `-superuser` parameter in export rules to determine how to handle clients presenting with user ID 0 depending on their security type. The following are valid options for the `-superuser` parameter:

- `any`
- `none`

  This is the default setting if you do not specify the `-superuser` parameter.

- `krb5`
- `ntlm`
- `sys`

There are two different ways how clients presenting with user ID 0 are handled, depending on the `-superuser` parameter configuration:

If the `-superuser` parameter and the client's security type…	Then the client…
Match	Gets superuser access with user ID 0.
Do not match	Gets access as the anonymous user with the user ID specified by the `-anon` parameter and its assigned permissions. This is regardless of whether the read-only or read-write parameter specifies the option `none`.

If a client presents with user ID 0 to access a volume with NTFS security style and the `-superuser` parameter is set to `none`, ONTAP uses the name mapping for the anonymous user to obtain the proper credentials.

Example

The export policy contains an export rule with the following parameters:

- `-protocol nfs3`
- `-clientmatch 10.1.16.0/255.255.255.0`
- `-rorule any`
- `-rwrule krb5,ntlm`
- `-anon 127`

Client #1 has the IP address 10.1.16.207, has user ID 746, sends an access request using the NFSv3 protocol, and authenticated with Kerberos v5.

Client #2 has the IP address 10.1.16.211, has user ID 0, sends an access request using the NFSv3 protocol, and authenticated with AUTH_SYS.

The client access protocol and IP address matches for both clients. The read-only parameter allows read-only access to all clients regardless of the security type they authenticated with. However, only client #1 gets read-write access because it used the approved security type Kerberos v5 to authenticate.

Client #2 does not get superuser access. Instead, it gets mapped to anonymous because the `-superuser` parameter is not specified. This means it defaults to `none` and automatically maps user ID 0 to anonymous. Client #2 also only gets read-only access because its security type did not match the read-write parameter.

Example

The export policy contains an export rule with the following parameters:

- `-protocol nfs3`
- `-clientmatch 10.1.16.0/255.255.255.0`
- `-rorule any`
- `-rwrule krb5,ntlm`
- `-superuser krb5`
- `-anon 0`

Client #1 has the IP address 10.1.16.207, has user ID 0, sends an access request using the NFSv3 protocol, and authenticated with Kerberos v5.

Client #2 has the IP address 10.1.16.211, has user ID 0, sends an access request using the NFSv3 protocol, and authenticated with AUTH_SYS.

The client access protocol and IP address matches for both clients. The read-only parameter allows read-only access to all clients regardless of the security type they authenticated with. However, only client #1 gets read-write access because it used the approved security type Kerberos v5 to authenticate. Client #2 does not get read-write access.

The export rule allows superuser access for clients with user ID 0. Client #1 gets superuser access because it matches the user ID and security type for the read-only and `-superuser` parameters. Client #2 does not get read-write or superuser access because its security type does not match the read-write parameter or the `-superuser` parameter. Instead, client #2 is mapped to the anonymous user, which in this case has the user ID 0.

How ONTAP uses export policy caches

To improve system performance, ONTAP uses local caches to store information such as host names and netgroups. This enables ONTAP to process export policy rules more quickly than retrieving the information from external sources. Understanding what the caches are and what they do can help you troubleshoot client access issues.

You configure export policies to control client access to NFS exports. Each export policy contains rules, and each rule contains parameters to match the rule to clients requesting access. Some of these parameters require ONTAP to contact an external source, such as DNS or NIS servers, to resolve objects such as domain names, host names, or netgroups.

These communications with external sources take a small amount of time. To increase performance, ONTAP reduces the amount of time it takes to resolve export policy rule objects by storing information locally on each node in several caches.

Cache name	Type of information stored
Access	Mappings of clients to corresponding export policies
Name	Mappings of UNIX user names to corresponding UNIX user IDs
ID	Mappings of UNIX user IDs to corresponding UNIX user IDs and extended UNIX group IDs
Host	Mappings of host names to corresponding IP addresses
Netgroup	Mappings of netgroups to corresponding IP addresses of members
Showmount	List of exported directories from SVM namespace

If you change information on the external name servers in your environment after ONTAP retrieved and stored it locally, the caches might now contain outdated information. Although ONTAP refreshes caches automatically after certain time periods, different caches have different expiration and refresh times and algorithms.

Another possible reason for caches to contain outdated information is when ONTAP attempts to refresh cached information but encounters a failure when attempting to communicate with name servers. If this happens, ONTAP continues to use the information currently stored in the local caches to prevent client disruption.

As a result, client access requests that are supposed to succeed might fail, and client access requests that are supposed to fail might succeed. You can view and manually flush some of the export policy caches when troubleshooting such client access issues.

How the access cache works

ONTAP uses an access cache to store the results of export policy rule evaluation for client access operations to a volume or qtree. This results in performance improvements because the information can be retrieved much faster from the access cache than going through the export policy rule evaluation process every time a client sends an I/O request.

Whenever an NFS client sends an I/O request to access data on a volume or qtree, ONTAP must evaluate each I/O request to determine whether to grant or deny the I/O request. This evaluation involves checking every export policy rule of the export policy associated with the volume or qtree. If the path to the volume or qtree involves crossing one or more junction points, this might require performing this check for multiple export policies along the path.

Note that this evaluation occurs for every I/O request sent from an NFS client, such as read, write, list, copy and other operations; not just for initial mount requests.

After ONTAP has identified the applicable export policy rules and decided whether to allow or deny the request, ONTAP then creates an entry in the access cache to store this information.

When an NFS client sends an I/O request, ONTAP notes the IP address of the client, the ID of the SVM, and the export policy associated with the target volume or qtree, and first checks the access cache for a matching entry. If a matching entry exists in the access cache, ONTAP uses the stored information to allow or deny the I/O request. If a matching entry does not exist, ONTAP then goes through the normal process of evaluating all applicable policy rules as explained above.

Access cache entries that are not actively used are not refreshed. This reduces unnecessary and wasteful communication with external name serves.

Retrieving the information from the access cache is much faster than going through the entire export policy rule evaluation process for every I/O request. Therefore, using the access cache greatly improves performance by reducing the overhead of client access checks.

How access cache parameters work

Several parameters control the refresh periods for entries in the access cache. Understanding how these parameters work enables you to modify them to tune the access cache and balance performance with how recent the stored information is.

The access cache stores entries consisting of one or more export rules that apply to clients attempting to access volumes or qtrees. These entries are stored for a certain amount of time before they are refreshed. The

refresh time is determined by access cache parameters and depends on the type of access cache entry.

You can specify access cache parameters for individual SVMs. This allows the parameters to differ according to SVM access requirements. Access cache entries that are not actively used are not refreshed, which reduces unnecessary and wasteful communication with external name serves.

Access cache entry type	Description	Refresh period in seconds
Positive entries	Access cache entries that have not resulted in access denial to clients.	Minimum: 300 Maximum: 86,400 Default: 3,600
Negative entries	Access cache entries that have resulted in access denial to clients.	Minimum: 60 Maximum: 86,400 Default: 3,600

Example

An NFS client attempts to access a volume on a cluster. ONTAP matches the client to an export policy rule and determines that the client gets access based on the export policy rule configuration. ONTAP stores the export policy rule in the access cache as a positive entry. By default, ONTAP keeps the positive entry in the access cache for one hour (3,600 seconds), and then automatically refreshes the entry to keep the information current.

To prevent the access cache from filling up unnecessarily, there is an additional parameter to clear existing access cache entries that have not been used for a certain time period to decide client access. This `-harvest-timeout` parameter has an allowed range of 60 through 2,592,000 seconds and a default setting of 86,400 seconds.

Remove an export policy from a qtree

If you decide you do not want a specific export policy assigned to a qtree any longer, you can remove the export policy by modifying the qtree to inherit the export policy of the containing volume instead. You can do this by using the `volume qtree modify` command with the `-export-policy` parameter and an empty name string ("").

Steps

1. To remove an export policy from a qtree, enter the following command:

   ```
   volume qtree modify -vserver vserver_name -qtree-path
   /vol/volume_name/qtree_name -export-policy ""
   ```

2. Verify that the qtree was modified accordingly:

   ```
   volume qtree show -qtree qtree_name -fields export-policy
   ```

Validate qtree IDs for qtree file operations

ONTAP can perform an optional additional validation of qtree IDs. This validation ensures

that client file operation requests use a valid qtree ID and that clients can only move files within the same qtree. You can enable or disable this validation by modifying the `-validate-qtree-export` parameter. This parameter is enabled by default.

About this task

This parameter is only effective when you have assigned an export policy directly to one or more qtrees on the storage virtual machine (SVM).

Steps

1. Set the privilege level to advanced:

   ```
   set -privilege advanced
   ```

2. Perform one of the following actions:

If you want qtree ID validation to be…	Enter the following command…
Enabled	`vserver nfs modify -vserver vserver_name -validate-qtree-export enabled`
Disabled	`vserver nfs modify -vserver vserver_name -validate-qtree-export disabled`

3. Return to the admin privilege level:

   ```
   set -privilege admin
   ```

Export policy restrictions and nested junctions for FlexVol volumes

If you configured export policies to set a less restrictive policy on a nested junction but a more restrictive policy on a higher level junction, access to the lower level junction might fail.

You should ensure that higher level junctions have less restrictive export policies than lower level junctions.

Using Kerberos with NFS for strong security

ONTAP support for Kerberos

Kerberos provides strong secure authentication for client/server applications. Authentication provides verification of user and process identities to a server. In the ONTAP environment, Kerberos provides authentication between storage virtual machines (SVMs) and NFS clients.

In ONTAP 9, the following Kerberos functionality is supported:

- Kerberos 5 authentication with integrity checking (krb5i)

Krb5i uses checksums to verify the integrity of each NFS message transferred between client and server. This is useful both for security reasons (for example, to ensure that data has not been tampered with) and for data integrity reasons (for example, to prevent data corruption when using NFS over unreliable networks).

- Kerberos 5 authentication with privacy checking (krb5p)

  Krb5p uses checksums to encrypt all the traffic between client and the server. This is more secure and also incurs more load.

- 128-bit and 256-bit AES encryption

  Advanced Encryption Standard (AES) is an encryption algorithm for securing electronic data. ONTAP now supports AES with 128-bit keys (AES-128) and AES with 256-bit keys (AES-256) encryption for Kerberos for stronger security.

- SVM-level Kerberos realm configurations

  SVM administrators can now create Kerberos realm configurations at the SVM level. This means that SVM administrators no longer have to rely on the cluster administrator for Kerberos realm configuration and can create individual Kerberos realm configurations in a multi-tenancy environment.

Requirements for configuring Kerberos with NFS

Before you configure Kerberos with NFS on your system, you must verify that certain items in your network and storage environment are properly configured.

> The steps to configure your environment depend on what version and type of client operating system, domain controller, Kerberos, DNS, etc., that you are using. Documenting all these variables is beyond the scope of this document. For more information, see the respective documentation for each component.
>
> For a detailed example of how to set up ONTAP and Kerberos 5 with NFSv3 and NFSv4 in an environment using Windows Server 2008 R2 Active Directory and Linux hosts, see technical report 4073.

The following items should be configured first:

Network environment requirements

- Kerberos

  You must have a working Kerberos setup with a key distribution center (KDC), such as Windows Active Directory based Kerberos or MIT Kerberos.

  NFS servers must use `nfs` as the primary component of their machine principal.

- Directory service

  You must use a secure directory service in your environment, such as Active Directory or OpenLDAP, that is configured to use LDAP over SSL/TLS.

- NTP

You must have a working time server running NTP. This is necessary to prevent Kerberos authentication failure due to time skew.

- Domain name resolution (DNS)

  Each UNIX client and each SVM LIF must have a proper service record (SRV) registered with the KDC under forward and reverse lookup zones. All participants must be properly resolvable via DNS.

- User accounts

  Each client must have a user account in the Kerberos realm. NFS servers must use “nfs” as the primary component of their machine principal.

NFS client requirements

- NFS

  Each client must be properly configured to communicate over the network using NFSv3 or NFSv4.

  Clients must support RFC1964 and RFC2203.

- Kerberos

  Each client must be properly configured to use Kerberos authentication, including the following details:

  - Encryption for TGS communication is enabled.

    AES-256 for strongest security.

  - The most secure encryption type for TGT communication is enabled.
  - The Kerberos realm and domain are configured correctly.
  - GSS is enabled.

    When using machine credentials:

  - Do not run `gssd` with the `-n` parameter.
  - Do not run `kinit` as the root user.

- Each client must use the most recent and updated operating system version.

  This provides the best compatibility and reliability for AES encryption with Kerberos.

- DNS

  Each client must be properly configured to use DNS for correct name resolution.

- NTP

  Each client must be synchronizing with the NTP server.

- Host and domain information

  Each client’s `/etc/hosts` and `/etc/resolv.conf` files must contain the correct host name and DNS information, respectively.

- Keytab files

  Each client must have a keytab file from the KDC. The realm must be in uppercase letters. The encryption type must be AES-256 for strongest security.

- Optional: For best performance, clients benefit from having at least two network interfaces: one for communicating with the local area network and one for communicating with the storage network.

Storage system requirements

- NFS license

  The storage system must have a valid NFS license installed.

- CIFS license

  The CIFS license is optional. It is only required for checking Windows credentials when using multiprotocol name mapping. It is not required in a strict UNIX-only environment.

- SVM

  You must have at least one SVM configured on the system.

- DNS on the SVM

  You must have configured DNS on each SVM.

- NFS server

  You must have configured NFS on the SVM.

- AES encryption

  For strongest security, you must configure the NFS server to allow only AES-256 encryption for Kerberos.

- SMB server

  If you are running a multiprotocol environment, you must have configured SMB on the SVM. The SMB server is required for multiprotocol name mapping.

- Volumes

  You must have a root volume and at least one data volume configured for use by the SVM.

- Root volume

  The root volume of the SVM must have the following configuration:

Name	Setting
Security style	UNIX
UID	root or ID 0

Name	Setting
GID	root or ID 0
UNIX permissions	777

In contrast to the root volume, data volumes can have either security style.

- UNIX groups

  The SVM must have the following UNIX groups configured:

Group name	Group ID
daemon	1
root	0
pcuser	65534 (created automatically by ONTAP when you create the SVM)

- UNIX users

  The SVM must have the following UNIX users configured:

User name	User ID	Primary group ID	Comment
nfs	500	0	Required for GSS INIT phase The first component of the NFS client user SPN is used as the user.
pcuser	65534	65534	Required for NFS and CIFS multiprotocol use Created and added to the pcuser group automatically by ONTAP when you create the SVM.
root	0	0	Required for mounting

  The nfs user is not required if a Kerberos-UNIX name mapping exists for the SPN of the NFS client user.

- Export policies and rules

  You must have configured export policies with the necessary export rules for the root and data volumes and qtrees. If all volumes of the SVM are accessed over Kerberos, you can set the export rule options `-rorule`, `-rwrule`, and `-superuser` for the root volume to `krb5` , `krb5i`, or `krb5p`.

- Kerberos-UNIX name mapping

  If you want the user identified by the NFS client user SPN to have root permissions, you must create a name mapping to root.

Related information

NetApp Technical Report 4073: Secure Unified Authentication

NetApp Interoperability Matrix Tool

System administration

Logical storage management

Specify the user ID domain for NFSv4

To specify the user ID domain, you can set the `-v4-id-domain` option.

About this task

By default, ONTAP uses the NIS domain for NFSv4 user ID mapping, if one is set. If an NIS domain is not set, the DNS domain is used. You might need to set the user ID domain if, for example, you have multiple user ID domains. The domain name must match the domain configuration on the domain controller. It is not required for NFSv3.

Step

1. Enter the following command:

```
vserver nfs modify -vserver vserver_name -v4-id-domain NIS_domain_name
```

Configure name services

How ONTAP name service switch configuration works

ONTAP stores name service configuration information in a table that is the equivalent of the `/etc/nsswitch.conf` file on UNIX systems. You must understand the function of the table and how ONTAP uses it so that you can configure it appropriately for your environment.

The ONTAP name service switch table determines which name service sources ONTAP consults in which order to retrieve information for a certain type of name service information. ONTAP maintains a separate name service switch table for each SVM.

Database types

The table stores a separate name service list for each of the following database types:

Database type	Defines name service sources for…	Valid sources are…
hosts	Converting host names to IP addresses	files, dns
group	Looking up user group information	files, nis, ldap
passwd	Looking up user information	files, nis, ldap
netgroup	Looking up netgroup information	files, nis, ldap
namemap	Mapping user names	files, ldap

Source types

The sources specify which name service source to use for retrieving the appropriate information.

Specify source type…	To look up information in…	Managed by the command families…

files	Local source files	`vserver services name-service unix-user` `vserver services name-service unix-group`
		
		`vserver services name-service netgroup`
		
		`vserver services name-service dns hosts`

nis	External NIS servers as specified in the NIS domain configuration of the SVM	`vserver services name-service nis-domain`
ldap	External LDAP servers as specified in the LDAP client configuration of the SVM	`vserver services name-service ldap`
dns	External DNS servers as specified in the DNS configuration of the SVM	`vserver services name-service dns`

Even if you plan to use NIS or LDAP for both data access and SVM administration authentication, you should still include `files` and configure local users as a fallback in case NIS or LDAP authentication fails.

Protocols used to access external sources

To access the servers for external sources, ONTAP uses the following protocols:

External name service source	Protocol used for access
NIS	UDP
DNS	UDP
LDAP	TCP

Example

The following example displays the name service switch configuration for the SVM svm_1:

```
cluster1::*> vserver services name-service ns-switch show -vserver svm_1
                                 Source
Vserver         Database         Order
--------------- ------------   ---------
svm_1           hosts          files,
                               dns
svm_1           group          files
svm_1           passwd         files
svm_1           netgroup       nis,
                               files
```

To look up IP addresses for hosts, ONTAP first consults local source files. If the query does not return any results, DNS servers are checked next.

To look up user or group information, ONTAP consults only local sources files. If the query does not return any results, the lookup fails.

To look up netgroup information, ONTAP first consults external NIS servers. If the query does not return any results, the local netgroup file is checked next.

There are no name service entries for name mapping in the table for the SVM svm_1. Therefore, ONTAP consults only local source files by default.

Related information

NetApp Technical Report 4668: Name Services Best Practices Guide

Use LDAP

LDAP Overview

An LDAP (Lightweight Directory Access Protocol) server enables you to centrally maintain user information. If you store your user database on an LDAP server in your environment, you can configure your storage system to look up user information in your existing LDAP database.

- Before configuring LDAP for ONTAP, you should verify that your site deployment meets best practices for LDAP server and client configuration. In particular, the following conditions must be met:
  - The domain name of the LDAP server must match the entry on the LDAP client.
  - The LDAP user password hash types supported by the LDAP server must include those supported by ONTAP:
    - CRYPT (all types) and SHA-1 (SHA, SSHA).
    - Beginning with ONTAP 9.8, SHA-2 hashes (SHA-256, SSH-384, SHA-512, SSHA-256, SSHA-384, and SSHA-512) are also supported.
  - If the LDAP server requires session security measures, you must configure them in the LDAP client.

    The following session security options are available:

    - LDAP signing (provides data integrity checking) and LDAP signing and sealing (provides data integrity checking and encryption)
    - START TLS
    - LDAPS (LDAP over TLS or SSL)
  - To enable signed and sealed LDAP queries, the following services must be configured:
    - LDAP servers must support the GSSAPI (Kerberos) SASL mechanism.
    - LDAP servers must have DNS A/AAAA records as well as PTR records set up on the DNS server.
    - Kerberos servers must have SRV records present on the DNS server.
  - To enable START TLS or LDAPS, the following points should be considered.
    - It is a NetApp best practice to use Start TLS rather than LDAPS.
    - If LDAPS is used, the LDAP server must be enabled for TLS or for SSL in ONTAP 9.5 and later. SSL is not supported in ONTAP 9.0-9.4.
    - A certificate server must already be configured in the domain.
  - To enable LDAP referral chasing (in ONTAP 9.5 and later), the following conditions must be satisfied:
    - Both domains should be configured with one of the following trust relationships:
      - Two-way
      - One-way, where the primary trusts the referral domain
      - Parent-child
    - DNS must be configured to resolve all referred server names.
    - Domain passwords should be same to authenticate when `--bind-as-cifs-server` set to true.

      The following configurations are not supported with LDAP referral chasing.

      - For all ONTAP versions:
      - LDAP clients on an admin SVM
      - For ONTAP 9.8 and earlier (they are supported in 9.9.1 and later):
      - LDAP signing and sealing (the `-session-security` option)
      - Encrypted TLS connections (the `-use-start-tls` option)
      - Communications over LDAPS port 636 (the `-use-ldaps-for-ad-ldap` option)

- Beginning with ONTAP 9.11.1, you can use LDAP fast bind for nsswitch authentication.
- You must enter an LDAP schema when configuring the LDAP client on the SVM.

  In most cases, one of the default ONTAP schemas will be appropriate. However, if the LDAP schema in your environment differs from these, you must create a new LDAP client schema for ONTAP before creating the LDAP client. Consult with your LDAP administrator about requirements for your environment.

- Using LDAP for host name resolution is not supported.

For additional information, see NetApp Technical Report 4835: How to Configure LDAP in ONTAP.

LDAP signing and sealing concepts

Beginning with ONTAP 9, you can configure signing and sealing to enable LDAP session security on queries to an Active Directory (AD) server. You must configure the NFS server security settings on the storage virtual machine (SVM) to correspond to those on the LDAP server.

Signing confirms the integrity of the LDAP payload data using secret key technology. Sealing encrypts the LDAP payload data to avoid transmitting sensitive information in clear text. An *LDAP Security Level* option indicates whether the LDAP traffic needs to be signed, signed and sealed, or neither. The default is `none`. test

LDAP signing and sealing on SMB traffic is enabled on the SVM with the `-session-security-for-ad-ldap` option to the `vserver cifs security modify` command.

LDAPS concepts

You must understand certain terms and concepts about how ONTAP secures LDAP communication. ONTAP can use START TLS or LDAPS for setting up authenticated sessions between Active Directory-integrated LDAP servers or UNIX-based LDAP servers.

Terminology

There are certain terms that you should understand about how ONTAP uses LDAPS to secure LDAP communication.

- **LDAP**

  (Lightweight Directory Access Protocol) A protocol for accessing and managing information directories. LDAP is used as an information directory for storing objects such as users, groups, and netgroups. LDAP also provides directory services that manage these objects and fulfill LDAP requests from LDAP clients.

- **SSL**

  (Secure Sockets Layer) A protocol developed for sending information securely over the Internet. It has been deprecated in favor of TLS. SSL is not supported in ONTAP 9.0-9.4.

- **TLS**

  (Transport Layer Security) An IETF standards track protocol that is based on the earlier SSL specifications. It is the successor to SSL.

- **LDAPS (LDAP over SSL or TLS)**

  A protocol that uses TLS or SSL to secure communication between LDAP clients and LDAP servers. The terms *LDAP over SSL* and *LDAP over TLS* are sometimes used interchangeably; TLS is supported by ONTAP 9 and later, SSL is supported by ONTAP 9.5 and later.

  - In ONTAP 9.5-9.8, LDAPS can only be enabled on port 636. To do so, use the `-use-ldaps-for-ad-ldap` parameter with the `vserver cifs security modify` command.
  - Beginning with ONTAP 9.9.1, LDAPS can be enabled on any port, although port 636 remains the default. To do so, set the `-ldaps-enabled` parameter to `true` and specify the desired `-port` parameter. For more information, see the `vserver services name-service ldap client create` man page

  It is a NetApp best practice to use Start TLS rather than LDAPS.

- **Start TLS**

  (Also known as *start_tls*, *STARTTLS*, and *StartTLS*) A mechanism to provide secure communication by using the TLS protocols.

  ONTAP uses STARTTLS for securing LDAP communication, and uses the default LDAP port (389) to communicate with the LDAP server. The LDAP server must be configured to allow connections over LDAP port 389; otherwise, LDAP TLS connections from the SVM to the LDAP server fail.

How ONTAP uses LDAPS

ONTAP supports TLS server authentication, which enables the SVM LDAP client to confirm the LDAP server's identity during the bind operation. TLS-enabled LDAP clients can use standard techniques of public-key cryptography to check that a server's certificate and public ID are valid and have been issued by a certificate authority (CA) listed in the client's list of trusted CAs.

LDAP supports STARTTLS to encrypt communications using TLS. STARTTLS begins as a plaintext connection over the standard LDAP port (389), and that connection is then upgraded to TLS.

ONTAP supports the following:

- LDAPS for SMB-related traffic between the Active Directory-integrated LDAP servers and the SVM
- LDAPS for LDAP traffic for name mapping and other UNIX information

  Either Active Directory-integrated LDAP servers or UNIX-based LDAP servers can be used to store information for LDAP name mapping and other UNIX information, such as users, groups, and netgroups.

- Self-signed root CA certificates

  When using an Active-Directory integrated LDAP, the self-signed root certificate is generated when the Windows Server Certificate Service is installed in the domain. When using an UNIX-based LDAP server for LDAP name mapping, the self-signed root certificate is generated and saved by using means appropriate to that LDAP application.

By default, LDAPS is disabled.

Enable LDAP RFC2307bis support

If you want to use LDAP and require the additional capability to use nested group memberships, you can configure ONTAP to enable LDAP RFC2307bis support.

What you'll need

You must have created a copy of one of the default LDAP client schemas that you want to use.

About this task

In LDAP client schemas, group objects use the memberUid attribute. This attribute can contain multiple values and lists the names of the users that belong to that group. In RFC2307bis enabled LDAP client schemas, group objects use the uniqueMember attribute. This attribute can contain the full distinguished name (DN) of another object in the LDAP directory. This enables you to use nested groups because groups can have other groups as members.

The user should not be a member of more than 256 groups including nested groups. ONTAP ignores any groups over the 256 group limit.

By default, RFC2307bis support is disabled.

RFC2307bis support is enabled automatically in ONTAP when an LDAP client is created with the MS-AD-BIS schema.

For additional information, see NetApp Technical Report 4835: How to Configure LDAP in ONTAP.

Steps

1. Set the privilege level to advanced:

   ```
   set -privilege advanced
   ```

2. Modify the copied RFC2307 LDAP client schema to enable RFC2307bis support:

   ```
   vserver services name-service ldap client schema modify -vserver vserver_name
   -schema schema-name -enable-rfc2307bis true
   ```

3. Modify the schema to match the object class supported in the LDAP server:

   ```
   vserver services name-service ldap client schema modify -vserver vserver-name
   -schema schema_name -group-of-unique-names-object-class object_class
   ```

4. Modify the schema to match the attribute name supported in the LDAP server:

   ```
   vserver services name-service ldap client schema modify -vserver vserver-name
   -schema schema_name -unique-member-attribute attribute_name
   ```

5. Return to the admin privilege level:

   ```
   set -privilege admin
   ```

Configuration options for LDAP directory searches

You can optimize LDAP directory searches, including user, group, and netgroup

information, by configuring the ONTAP LDAP client to connect to LDAP servers in the most appropriate way for your environment. You need to understand when the default LDAP base and scope search values suffice and which parameters to specify when custom values are more appropriate.

LDAP client search options for user, group, and netgroup information can help avoid failed LDAP queries, and therefore failed client access to storage systems. They also help ensure that the searches are as efficient as possible to avoid client performance issues.

Default base and scope search values

The LDAP base is the default base DN that the LDAP client uses to perform LDAP queries. All searches, including user, group, and netgroup searches, are done using the base DN. This option is appropriate when your LDAP directory is relatively small and all relevant entries are located in the same DN.

If you do not specify a custom base DN, the default is `root`. This means that each query searches the entire directory. Although this maximizes the chances of success of the LDAP query, it can be inefficient and result in significantly decreased performance with large LDAP directories.

The LDAP base scope is the default search scope that the LDAP client uses to perform LDAP queries. All searches, including user, group, and netgroup searches, are done using the base scope. It determines whether the LDAP query searches only the named entry, entries one level below the DN, or the entire subtree below the DN.

If you do not specify a custom base scope, the default is `subtree`. This means that each query searches the entire subtree below the DN. Although this maximizes the chances of success of the LDAP query, it can be inefficient and result in significantly decreased performance with large LDAP directories.

Custom base and scope search values

Optionally, you can specify separate base and scope values for user, group, and netgroup searches. Limiting the search base and scope of queries this way can significantly improve performance because it limits the search to a smaller subsection of the LDAP directory.

If you specify custom base and scope values, they override the general default search base and scope for user, group, and netgroup searches. The parameters to specify custom base and scope values are available at the advanced privilege level.

LDAP client parameter…	Specifies custom…
`-base-dn`	Base DN for all LDAP searchesMultiple values can be entered if needed (for example, if LDAP referral chasing is enabled in ONTAP 9.5 and later releases).
`-base-scope`	Base scope for all LDAP searches
`-user-dn`	Base DNs for all LDAP user searchesThis parameter also applies to user name-mapping searches.
`-user-scope`	Base scope for all LDAP user searches This parameter also applies to user name-mapping searches.

`-group-dn`	Base DNs for all LDAP group searches
`-group-scope`	Base scope for all LDAP group searches
`-netgroup-dn`	Base DNs for all LDAP netgroup searches
`-netgroup-scope`	Base scope for all LDAP netgroup searches

Multiple custom base DN values

If your LDAP directory structure is more complex, it might be necessary for you to specify multiple base DNs to search multiple parts of your LDAP directory for certain information. You can specify multiple DNs for the user, group, and netgroup DN parameters by separating them with a semicolon (;) and enclosing the entire DN search list with double quotes ("). If a DN contains a semicolon, you must add an escape character (\) immediately before the semicolon in the DN.

Note that the scope applies to the entire list of DNs specified for the corresponding parameter. For example, if you specify a list of three different user DNs and subtree for the user scope, then LDAP user searches search the entire subtree for each of the three specified DNs.

Beginning with ONTAP 9.5, you can also specify LDAP *referral chasing*, which allows the ONTAP LDAP client to refer look-up requests to other LDAP servers if an LDAP referral response is not returned by the primary LDAP server. The client uses that referral data to retrieve the target object from the server described in the referral data. To search for objects present in the referred LDAP servers, the base-dn of the referred objects can be added to the base-dn as part of LDAP client configuration. However, referred objects are only looked up when referral chasing is enabled (using the `-referral-enabled true` option) during LDAP client creation or modification.

Improve performance of LDAP directory netgroup-by-host searches

If your LDAP environment is configured to allow netgroup-by-host searches, you can configure ONTAP to take advantage of this and perform netgroup-by-host searches. This can significantly speed up netgroup searches and reduce possible NFS client access issues due to latency during netgroup searches.

What you'll need

Your LDAP directory must contain a `netgroup.byhost` map.

Your DNS servers should contain both forward (A) and reverse (PTR) lookup records for NFS clients.

When you specify IPv6 addresses in netgroups, you must always shorten and compress each address as specified in RFC 5952.

About this task

NIS servers store netgroup information in three separate maps called `netgroup`, `netgroup.byuser`, and `netgroup.byhost`. The purpose of the `netgroup.byuser` and `netgroup.byhost` maps is to speed up netgroup searches. ONTAP can perform netgroup-by-host searches on NIS servers for improved mount response times.

By default, LDAP directories do not have such a `netgroup.byhost` map like NIS servers. It is possible,

though, with the help of third-party tools, to import a NIS `netgroup.byhost` map into LDAP directories to enable fast netgroup-by-host searches. If you have configured your LDAP environment to allow netgroup-by-host searches, you can configure the ONTAP LDAP client with the `netgroup.byhost` map name, DN, and search scope for faster netgroup-by-host searches.

Receiving the results for netgroup-by-host searches faster enables ONTAP to process export rules faster when NFS clients request access to exports. This reduces the chance of delayed access due to netgroup search latency issues.

Steps

1. Obtain the exact full distinguished name of the NIS `netgroup.byhost` map you imported into your LDAP directory.

   The map DN can vary depending on the third-party tool you used for import. For best performance, you should specify the exact map DN.

2. Set the privilege level to advanced: `set -privilege advanced`
3. Enable netgroup-by-host searches in the LDAP client configuration of the storage virtual machine (SVM): `vserver services name-service ldap client modify -vserver vserver_name -client -config config_name -is-netgroup-byhost-enabled true -netgroup-byhost-dn netgroup-by-host_map_distinguished_name -netgroup-byhost-scope netgroup-by-host_search_scope`

   `-is-netgroup-byhost-enabled` {`true`|`false`} enables or disables netgroup-by-host search for LDAP directories. The default is `false`.

   `-netgroup-byhost-dn` `netgroup-by-host_map_distinguished_name` specifies the distinguished name of the `netgroup.byhost` map in the LDAP directory. It overrides the base DN for netgroup-by-host searches. If you do not specify this parameter, ONTAP uses the base DN instead.

   `-netgroup-byhost-scope` {`base`|`onelevel`|`subtree`} specifies the search scope for netgroup-by-host searches. If you do not specify this parameter, the default is `subtree`.

   If the LDAP client configuration does not exist yet, you can enable netgroup-by-host searches by specifying these parameters when creating a new LDAP client configuration using the `vserver services name-service ldap client create` command.

   > Beginning with ONTAP 9.2, the field `-ldap-servers` replaces the field `-servers`. This new field can take either a hostname or an IP address for the LDAP server.

4. Return to the admin privilege level: `set -privilege admin`

Example

The following command modifies the existing LDAP client configuration named “ldap_corp” to enable netgroup-by-host searches using the `netgroup.byhost` map named “nisMapName="netgroup.byhost",dc=corp,dc=example,dc=com” and the default search scope `subtree`:

```
cluster1::*> vserver services name-service ldap client modify -vserver vs1
-client-config ldap_corp -is-netgroup-byhost-enabled true -netgroup-byhost
-dn nisMapName="netgroup.byhost",dc=corp,dc=example,dc=com
```

After you finish

The `netgroup.byhost` and `netgroup` maps in the directory must be kept in sync at all times to avoid client access issues.

Related information

IETF RFC 5952: A Recommendation for IPv6 Address Text Representation

Use LDAP fast bind for nsswitch authentication

Beginning with ONTAP 9.11.1, you can take advantage of LDAP *fast bind* functionality (also known as *concurrent bind*) for faster and simpler client authentication requests. To use this functionality, the LDAP server must support fast bind functionality.

About this task

Without fast bind, ONTAP uses LDAP simple bind to authenticate admin users with the LDAP server. With this authentication method, ONTAP sends a user or group name to the LDAP server, receives the stored hash password, and compares the server hash code with the hash passcode generated locally from the user password. If they are identical, ONTAP grants login permission.

With fast bind functionality, ONTAP sends only user credentials (user name and password) to the LDAP server through a secure connection. The LDAP server then validates these credentials and instructs ONTAP to grant login permissions.

One advantage of fast bind is that there is no need for ONTAP to support every new hashing algorithm supported by LDAP servers, because password hashing is performed by the LDAP server.

Learn about using fast bind.

You can use existing LDAP client configurations for LDAP fast bind. However, it is strongly recommended that the LDAP client be configured for TLS or LDAPs; otherwise, the password is sent over the wire in plain text.

To enable LDAP fast bind in an ONTAP environment, you must satisfy these requirements:

- ONTAP admin users must be configured on an LDAP server that supports fast bind.
- The ONTAP SVM must be configured for LDAP in the name services switch (nsswitch) database.
- ONTAP admin user and group accounts must be configured for nsswitch authentication using fast bind.

Steps

1. Confirm with your LDAP administrator that LDAP fast bind is supported on the LDAP server.
2. Ensure that ONTAP admin user credentials are configured on the LDAP server.
3. Verify that the admin or data SVM is configured correctly for LDAP fast bind.
   a. To confirm that the LDAP fast bind server is listed in the LDAP client configuration, enter:

      ```
      vserver services name-service ldap client show
      ```

      Learn about LDAP client configuration.

   b. To confirm that `ldap` is one of the configured sources for the nsswitch `passwd` database, enter:

      ```
      vserver services name-service ns-switch show
      ```

Learn about nsswitch configuration.

4. Ensure that admin users are authenticating with nsswitch and that LDAP fast bind authentication is enabled in their accounts.
   - For existing users, enter `security login modify` and verify the following parameter settings:

     `-authentication-method nsswitch`

     `-is-ldap-fastbind true`

   - For new admin users, see Enable LDAP or NIS account access.

Display LDAP statistics

Beginning with ONTAP 9.2, you can display LDAP statistics for storage virtual machines (SVMs) on a storage system to monitor the performance and diagnose issues.

What you'll need

- You must have configured an LDAP client on the SVM.
- You must have identified LDAP objects from which you can view data.

Step

1. View the performance data for counter objects:

   `statistics show`

Examples

The following example shows the performance data for object `secd_external_service_op`:

```
cluster::*> statistics show -vserver vserverName -object
secd_external_service_op -instance "vserverName:LDAP (NIS & Name
Mapping):GetUserInfoFromName:1.1.1.1"

Object: secd_external_service_op
Instance: vserverName:LDAP (NIS & Name
Mapping):GetUserInfoFromName:1.1.1.1
Start-time: 4/13/2016 22:15:38
End-time: 4/13/2016 22:15:38
Scope: vserverName
Counter                              Value
-------------------------------- --------------------------------
instance_name                        vserverName:LDAP (NIS & Name
                                     Mapping):GetUserInfoFromName:
                                     1.1.1.1
last_modified_time                   1460610787
node_name                            nodeName
num_not_found_responses              1
num_request_failures                 1
num_requests_sent                    1
num_responses_received               1
num_successful_responses             0
num_timeouts                         0
operation                            GetUserInfoFromName
process_name                         secd
request_latency                      52131us
```

Configure name mappings

Configure name mappings overview

ONTAP uses name mapping to map SMB identities to UNIX identities, Kerberos identities to UNIX identities, and UNIX identities to SMB identities. It needs this information to obtain user credentials and provide proper file access regardless of whether they are connecting from an NFS client or a SMB client.

There are two exceptions where you do not have to use name mapping:

- You configure a pure UNIX environment and do not plan to use SMB access or NTFS security style on volumes.
- You configure the default user to be used instead.

  In this scenario, name mapping is not required because instead of mapping every individual client credential all client credentials are mapped to the same default user.

Note that you can use name mapping only for users, not for groups.

However, you can map a group of individual users to a specific user. For example, you can map all AD users that start or end with the word SALES to a specific UNIX user and to the user's UID.

How name mapping works

When ONTAP has to map credentials for a user, it first checks the local name mapping database and LDAP server for an existing mapping. Whether it checks one or both and in which order is determined by the name service configuration of the SVM.

- For Windows to UNIX mapping

  If no mapping is found, ONTAP checks whether the lowercase Windows user name is a valid user name in the UNIX domain. If this does not work, it uses the default UNIX user provided that it is configured. If the default UNIX user is not configured and ONTAP cannot obtain a mapping this way either, mapping fails and an error is returned.

- For UNIX to Windows mapping

  If no mapping is found, ONTAP tries to find a Windows account that matches the UNIX name in the SMB domain. If this does not work, it uses the default SMB user, provided that it is configured. If the default SMB user is not configured and ONTAP cannot obtain a mapping this way either, mapping fails and an error is returned.

Machine accounts are mapped to the specified default UNIX user by default. If no default UNIX user is specified, machine account mappings fail.

- Beginning with ONTAP 9.5, you can map machine accounts to users other than the default UNIX user.
- In ONTAP 9.4 and earlier, you cannot map machine accounts to other users.

  Even if name mappings for machine accounts are defined, the mappings are ignored.

Multidomain searches for UNIX user to Windows user name mappings

ONTAP supports multidomain searches when mapping UNIX users to Windows users. All discovered trusted domains are searched for matches to the replacement pattern until a matching result is returned. Alternatively, you can configure a list of preferred trusted domains, which is used instead of the discovered trusted domain list and is searched in order until a matching result is returned.

How domain trusts affect UNIX user to Windows user name mapping searches

To understand how multidomain user name mapping works, you must understand how domain trusts work with ONTAP. Active Directory trust relationships with the SMB server's home domain can be a bidirectional trust or can be one of two types of unidirectional trusts, either an inbound trust or an outbound trust. The home domain is the domain to which the SMB server on the SVM belongs.

- *Bidirectional trust*

  With bidirectional trusts, both domains trust each other. If the SMB server's home domain has a bidirectional trust with another domain, the home domain can authenticate and authorize a user belonging to the trusted domain and vice versa.

UNIX user to Windows user name mapping searches can be performed only on domains with bidirectional trusts between the home domain and the other domain.

- *Outbound trust*

  With an outbound trust, the home domain trusts the other domain. In this case, the home domain can authenticate and authorize a user belonging to the outbound trusted domain.

  A domain with an outbound trust with the home domain is *not* searched when performing UNIX user to Windows user name mapping searches.

- *Inbound trust*

  With an inbound trust, the other domain trusts the SMB server's home domain. In this case, the home domain cannot authenticate or authorize a user belonging to the inbound trusted domain.

  A domain with an inbound trust with the home domain is *not* searched when performing UNIX user to Windows user name mapping searches.

How wildcards (*) are used to configure multidomain searches for name mapping

Multidomain name mapping searches are facilitated by the use of wildcards in the domain section of the Windows user name. The following table illustrates how to use wildcards in the domain part of a name mapping entry to enable multidomain searches:

Pattern	Replacement	Result
root	*\\administrator	The UNIX user "root" is mapped to the user named "administrator". All trusted domains are searched in order until the first matching user named "administrator" is found.
*	**	Valid UNIX users are mapped to the corresponding Windows users. All trusted domains are searched in order until the first matching user with that name is found. The pattern ** is only valid for name mapping from UNIX to Windows, not the other way around.

How multidomain name searches are performed

You can choose one of two methods for determining the list of trusted domains used for multidomain name searches:

- Use the automatically discovered bidirectional trust list compiled by ONTAP
- Use the preferred trusted domain list that you compile

If a UNIX user is mapped to a Windows user with a wildcard used for the domain section of the user name, the Windows user is looked up in all the trusted domains as follows:

- If a preferred trusted-domain list is configured, the mapped Windows user is looked up in this search list only, in order.
- If a preferred list of trusted domains is not configured, then the Windows user is looked up in all the bidirectional trusted domains of the home domain.
- If there are no bidirectionally trusted domains for the home domain, the user is looked up in the home domain.

If a UNIX user is mapped to a Windows user without a domain section in the user name, the Windows user is looked up in the home domain.

Name mapping conversion rules

An ONTAP system keeps a set of conversion rules for each SVM. Each rule consists of two pieces: a *pattern* and a *replacement*. Conversions start at the beginning of the appropriate list and perform a substitution based on the first matching rule. The pattern is a UNIX-style regular expression. The replacement is a string containing escape sequences representing subexpressions from the pattern, as in the UNIX `sed` program.

Create a name mapping

You can use the `vserver name-mapping create` command to create a name mapping. You use name mappings to enable Windows users to access UNIX security style volumes and the reverse.

About this task

For each SVM, ONTAP supports up to 12,500 name mappings for each direction.

Step

1. Create a name mapping:

   ```
   vserver name-mapping create -vserver vserver_name -direction {krb-unix|win-unix|unix-win} -position integer -pattern text -replacement text
   ```

   > The `-pattern` and `-replacement` statements can be formulated as regular expressions. You can also use the `-replacement` statement to explicitly deny a mapping to the user by using the null replacement string `" "` (the space character). See the `vserver name-mapping create` man page for details.

   When Windows-to-UNIX mappings are created, any SMB clients that have open connections to the ONTAP system at the time the new mappings are created must log out and log back in to see the new mappings.

Examples

The following command creates a name mapping on the SVM named vs1. The mapping is a mapping from UNIX to Windows at position 1 in the priority list. The mapping maps the UNIX user johnd to the Windows user ENG\JohnDoe.

```
vs1::> vserver name-mapping create -vserver vs1 -direction unix-win
-position 1 -pattern johnd
-replacement "ENG\\JohnDoe"
```

The following command creates another name mapping on the SVM named vs1. The mapping is a mapping from Windows to UNIX at position 1 in the priority list. Here the pattern and replacement include regular expressions. The mapping maps every CIFS user in the domain ENG to users in the LDAP domain associated with the SVM.

```
vs1::> vserver name-mapping create -vserver vs1 -direction win-unix
-position 1 -pattern "ENG\\(.+)"
-replacement "\1"
```

The following command creates another name mapping on the SVM named vs1. Here the pattern includes “$” as an element in the Windows user name that must be escaped. The mapping maps the windows user ENG\john$ops to UNIX user john_ops.

```
vs1::> vserver name-mapping create -direction win-unix -position 1
-pattern ENG\\john\$ops
-replacement john_ops
```

Configure the default user

You can configure a default user to use if all other mapping attempts fail for a user, or if you do not want to map individual users between UNIX and Windows. Alternatively, if you want authentication of non-mapped users to fail, you should not configure a default user.

About this task

For CIFS authentication, if you do not want to map each Windows user to an individual UNIX user, you can instead specify a default UNIX user.

For NFS authentication, if you do not want to map each UNIX user to an individual Windows user, you can instead specify a default Windows user.

Step

1. Perform one of the following actions:

If you want to…	Enter the following command…
Configure the default UNIX user	`vserver cifs options modify -default-unix-user user_name`
Configure the default Windows user	`vserver nfs modify -default-win-user user_name`

Commands for managing name mappings

There are specific ONTAP commands for managing name mappings.

If you want to…	Use this command…
Create a name mapping	`vserver name-mapping create`
Insert a name mapping at a specific position	`vserver name-mapping insert`
Display name mappings	`vserver name-mapping show`
Exchange the position of two name mappings NOTE: A swap is not allowed when name-mapping is configured with an ip-qualifier entry.	`vserver name-mapping swap`
Modify a name mapping	`vserver name-mapping modify`
Delete a name mapping	`vserver name-mapping delete`
Validate the correct name mapping	`vserver security file-directory show-effective-permissions -vserver vs1 -win-user-name user1 -path / -share-name sh1`

See the man page for each command for more information.

Enable access for Windows NFS clients

ONTAP supports file access from Windows NFSv3 clients. This means that clients running Windows operating systems with NFSv3 support can now access files on NFSv3 exports on the cluster. To successfully use this functionality, you must properly configure the storage virtual machine (SVM) and be aware of certain requirements and limitations.

What you'll need

NFSv3 must be enabled on the SVM.

About this task

By default, Windows NFSv3 client support is disabled.

Windows NFSv3 clients do not support the network status monitor (NSM) protocol. As a result, Windows NFSv3 client sessions might experience disruptions during storage failover and volume move operations.

Steps

1. Enable Windows NFSv3 client support:

```
vserver nfs modify -vserver vserver_name -v3-ms-dos-client enabled
```

2. On all SVMs that support Windows NFSv3 clients, disable the `-enable-ejukebox` and `-v3 -connection-drop` parameters: `vserver nfs modify -vserver vserver_name -enable -ejukebox false -v3-connection-drop disabled`

   Windows NFSv3 clients can now mount exports on the storage system.

3. Ensure that each Windows NFSv3 client uses hard mounts by specifying the `-o mtype=hard` option.

   This is required to ensure reliable mounts.

   ```
   mount -o mtype=hard \\10.53.33.10\vol\vol1 z:\
   ```

Enable the display of NFS exports on NFS clients

NFS clients can use the `showmount -e` command to see a list of exports available from an ONTAP NFS server. This can help users identify the file system they want to mount.

Beginning with ONTAP 9.2, ONTAP allows NFS clients to view the export list by default. In earlier releases, the `showmount` option of the `vserver nfs modify` command must be enabled explicitly. For viewing the export list, NFSv3 should be enabled on the SVM.

Example

The following command shows the showmount feature on the SVM named vs1:

```
clusterl : : > vserver nfs show -vserver vs1 -fields showmount
vserver showmount
------- ---------
vs1    enabled
```

The following command executed on an NFS client displays the list of exports on an NFS server with the IP address 10.63.21.9:

```
showmount -e 10.63.21.9
Export list for 10.63.21.9:
/unix        (everyone)
/unix/unixl  (everyone)
/unix/unix2  (everyone)
/            (everyone)
```

Manage file access using NFS

Enable or disable NFSv3

You can enable or disable NFSv3 by modifying the `-v3` option. This allows file access for clients using the NFSv3 protocol. By default, NFSv3 is enabled.

Step

1. Perform one of the following actions:

If you want to…	Enter the command…
Enable NFSv3	`vserver nfs modify -vserver vserver_name -v3 enabled`
Disable NFSv3	`vserver nfs modify -vserver vserver_name -v3 disabled`

Enable or disable NFSv4.0

You can enable or disable NFSv4.0 by modifying the `-v4.0` option. This allows file access for clients using the NFSv4.0 protocol. In ONTAP 9.9.1, NFSv4.0 is enabled by default; in earlier releases, it is disabled by default.

Step

1. Perform one of the following actions:

If you want to…	Enter the following command…
Enable NFSv4.0	`vserver nfs modify -vserver vserver_name -v4.0 enabled`
Disable NFSv4.0	`vserver nfs modify -vserver vserver_name -v4.0 disabled`

Enable or disable NFSv4.1

You can enable or disable NFSv4.1 by modifying the `-v4.1` option. This allows file access for clients using the NFSv4.1 protocol. In ONTAP 9.9.1, NFSv4.1 is enabled by default; in earlier releases, it is disabled by default.

Step

1. Perform one of the following actions:

If you want to…	Enter the following command…
Enable NFSv4.1	`vserver nfs modify -vserver vserver_name -v4.1 enabled`
Disable NFSv4.1	`vserver nfs modify -vserver vserver_name -v4.1 disabled`

Enable or disable pNFS

pNFS improves performance by allowing NFS clients to perform read/write operations on storage devices directly and in parallel, bypassing the NFS server as a potential bottleneck. To enable or disable pNFS (parallel NFS), you can modify the `-v4.1-pnfs` option.

If the ONTAP release is…	The pNFS default is…
9.8 or later	disabled
9.7 or earlier	enabled

What you'll need

NFSv4.1 support is required to be able to use pNFS.

If you want to enable pNFS, you must first disable NFS referrals. They cannot both be enabled at the same time.

If you use pNFS with Kerberos on SVMs, you must enable Kerberos on every LIF on the SVM.

Step

1. Perform one of the following actions:

If you want to…	Enter the command…
Enable pNFS	`vserver nfs modify -vserver vserver_name -v4.1-pnfs enabled`
Disable pNFS	`vserver nfs modify -vserver vserver_name -v4.1-pnfs disabled`

Control NFS access over TCP and UDP

You can enable or disable NFS access to storage virtual machines (SVMs) over TCP and UDP by modifying the `-tcp` and `-udp` parameters, respectively. This enables you to control whether NFS clients can access data over TCP or UDP in your environment.

About this task

These parameters only apply to NFS. They do not affect auxiliary protocols. For example, if NFS over TCP is disabled, mount operations over TCP still succeed. To completely block TCP or UDP traffic, you can use export policy rules.

> You must turn off the SnapDiff RPC Server before you disable TCP for NFS to avoid a command failed error. You can disable TCP by using the command `vserver snapdiff-rpc-server off -vserver vserver name`.

Step

1. Perform one of the following actions:

If you want NFS access to be…	Enter the command…
Enabled over TCP	`vserver nfs modify -vserver vserver_name -tcp enabled`
Disabled over TCP	`vserver nfs modify -vserver vserver_name -tcp disabled`
Enabled over UDP	`vserver nfs modify -vserver vserver_name -udp enabled`
Disabled over UDP	`vserver nfs modify -vserver vserver_name -udp disabled`

Control NFS requests from nonreserved ports

You can reject NFS mount requests from nonreserved ports by enabling the `-mount -rootonly` option. To reject all NFS requests from nonreserved ports, you can enable the `-nfs-rootonly` option.

About this task

By default, the option `-mount-rootonly` is `enabled`.

By default, the option `-nfs-rootonly` is `disabled`.

These options do not apply to the NULL procedure.

Step

1. Perform one of the following actions:

If you want to…	Enter the command…
Allow NFS mount requests from nonreserved ports	`vserver nfs modify -vserver vserver_name -mount -rootonly disabled`
Reject NFS mount requests from nonreserved ports	`vserver nfs modify -vserver vserver_name -mount -rootonly enabled`
Allow all NFS requests from nonreserved ports	`vserver nfs modify -vserver vserver_name -nfs -rootonly disabled`
Reject all NFS requests from nonreserved ports	`vserver nfs modify -vserver vserver_name -nfs -rootonly enabled`

Handle NFS access to NTFS volumes or qtrees for unknown UNIX users

If ONTAP cannot identify UNIX users attempting to connect to volumes or qtrees with NTFS security style, it therefore cannot explicitly map the user to a Windows user. You

can configure ONTAP to either deny access to such users for stricter security or map them to a default Windows user to ensure a minimum level of access for all users.

What you'll need

A default Windows user must be configured if you want to enable this option.

About this task

If a UNIX user tries to access volumes or qtrees with NTFS security style, the UNIX user must first be mapped to a Windows user so that ONTAP can properly evaluate the NTFS permissions. However, if ONTAP cannot look up the name of the UNIX user in the configured user information name service sources, it cannot explicitly map the UNIX user to a specific Windows user. You can decide how to handle such unknown UNIX users in the following ways:

- Deny access to unknown UNIX users.

  This enforces stricter security by requiring explicit mapping for all UNIX users to gain access to NTFS volumes or qtrees.

- Map unknown UNIX users to a default Windows user.

  This provides less security but more convenience by ensuring that all users get a minimum level of access to NTFS volumes or qtrees through a default Windows user.

Steps

1. Set the privilege level to advanced:

   ```
   set -privilege advanced
   ```

2. Perform one of the following actions:

If you want the default Windows user for unknown UNIX users…	Enter the command…
Enabled	`vserver nfs modify -vserver vserver_name -map -unknown-uid-to-default-windows-user enabled`
Disabled	`vserver nfs modify -vserver vserver_name -map -unknown-uid-to-default-windows-user disabled`

3. Return to the admin privilege level:

   ```
   set -privilege admin
   ```

Considerations for clients that mount NFS exports using a nonreserved port

The `-mount-rootonly` option must be disabled on a storage system that must support clients that mount NFS exports using a nonreserved port even when the user is logged in as root. Such clients include Hummingbird clients and Solaris NFS/IPv6 clients.

If the `-mount-rootonly` option is enabled, ONTAP does not allow NFS clients that use nonreserved ports,

meaning ports with numbers higher than 1,023, to mount NFS exports.

Perform stricter access checking for netgroups by verifying domains

By default, ONTAP performs an additional verification when evaluating client access for a netgroup. The additional check ensures that the client's domain matches the domain configuration of the storage virtual machine (SVM). Otherwise, ONTAP denies client access.

About this task

When ONTAP evaluates export policy rules for client access and an export policy rule contains a netgroup, ONTAP must determine whether a client's IP address belongs to the netgroup. For this purpose, ONTAP converts the client's IP address to a host name using DNS and obtains a fully qualified domain name (FQDN).

If the netgroup file only lists a short name for the host and the short name for the host exists in multiple domains, it is possible for a client from a different domain to obtain access without this check.

To prevent this, ONTAP compares the domain that was returned from DNS for the host against the list of DNS domain names configured for the SVM. If it matches, access is allowed. If it does not match, access is denied.

This verification is enabled by default. You can manage it by modifying the `-netgroup-dns-domain -search` parameter, which is available at the advanced privilege level.

Steps

1. Set the privilege level to advanced:

   ```
   set -privilege advanced
   ```

2. Perform the desired action:

If you want domain verification for netgroups to be…	Enter…
Enabled	`vserver nfs modify -vserver vserver_name -netgroup-dns-domain -search enabled`
Disabled	`vserver nfs modify -vserver vserver_name -netgroup-dns-domain -search disabled`

3. Set the privilege level to admin:

   ```
   set -privilege admin
   ```

Modify ports used for NFSv3 services

The NFS server on the storage system uses services such as mount daemon and Network Lock Manager to communicate with NFS clients over specific default network ports. In most NFS environments the default ports work correctly and do not require

modification, but if you want to use different NFS network ports in your NFSv3 environment, you can do so.

What you'll need

Changing NFS ports on the storage system requires that all NFS clients reconnect to the system, so you should communicate this information to your users in advance of making the change.

About this task

You can set the ports used by the NFS mount daemon, Network Lock Manager, Network Status Monitor, and NFS quota daemon services for each storage virtual machine (SVM). The port number change affects NFS clients accessing data over both TCP and UDP.

Ports for NFSv4 and NFSv4.1 cannot be changed.

Steps

1. Set the privilege level to advanced:

   ```
   set -privilege advanced
   ```

2. Disable access to NFS:

   ```
   vserver nfs modify -vserver vserver_name -access false
   ```

3. Set the NFS port for the specific NFS service:

   ```
   vserver nfs modify -vserver vserver_namenfs_port_parameterport_number
   ```

NFS port parameter	Description	Default port
`-mountd-port`	NFS mount daemon	635
`-nlm-port`	Network Lock Manager	4045
`-nsm-port`	Network Status Monitor	4046
`-rquotad-port`	NFS quota daemon	4049

   Besides the default port, the allowed range of port numbers is 1024 through 65535. Each NFS service must use a unique port.

4. Enable access to NFS:

   ```
   vserver nfs modify -vserver vserver_name -access true
   ```

5. Use the `network connections listening show` command to verify the port number changes.
6. Return to the admin privilege level:

   ```
   set -privilege admin
   ```

Example

The following commands set the NFS Mount Daemon port to 1113 on the SVM named vs1:

```
vs1::> set -privilege advanced
Warning: These advanced commands are potentially dangerous; use
         them only when directed to do so by NetApp personnel.
Do you want to continue? {y|n}: y

vs1::*> vserver nfs modify -vserver vs1 -access false

vs1::*> vserver nfs modify -vserver vs1 -mountd-port 1113

vs1::*> vserver nfs modify -vserver vs1 -access true

vs1::*> network connections listening show
Vserver Name     Interface Name:Local Port        Protocol/Service
---------------- -------------------------------  ----------------
Node: cluster1-01
Cluster          cluster1-01_clus_1:7700          TCP/ctlopcp
vs1              data1:4046                       TCP/sm
vs1              data1:4046                       UDP/sm
vs1              data1:4045                       TCP/nlm-v4
vs1              data1:4045                       UDP/nlm-v4
vs1              data1:1113                       TCP/mount
vs1              data1:1113                       UDP/mount
...
vs1::*> set -privilege admin
```

Commands for managing NFS servers

There are specific ONTAP commands for managing NFS servers.

If you want to…	Use this command…
Create an NFS server	`vserver nfs create`
Display NFS servers	`vserver nfs show`
Modify an NFS server	`vserver nfs modify`
Delete an NFS server	`vserver nfs delete`

Hide the `.snapshot` directory listing under NFSv3 mount points Explicit access to the `.snapshot` directory will still be allowed even if the option is enabled.	`vserver nfs` commands with the `-v3-hide-snapshot` option enabled

See the man page for each command for more information.

Troubleshoot name service issues

When clients experience access failures due to name service issues, you can use the `vserver services name-service getxxbyyy` command family to manually perform various name service lookups and examine the details and results of the lookup to help with troubleshooting.

About this task

- For each command, you can specify the following:
  - Name of the node or storage virtual machine (SVM) to perform the lookup on.

    This enables you to test name service lookups for a specific node or SVM to narrow the search for a potential name service configuration issue.
  - Whether to show the source used for the lookup.

    This enables you to check whether the correct source was used.
- ONTAP selects the service for performing the lookup based on the configured name service switch order.
- These commands are available at the advanced privilege level.

Steps

1. Perform one of the following actions:

To retrieve the…	Use the command…
IP address of a host name	`vserver services name-service getxxbyyy getaddrinfo` `vserver services name-service getxxbyyy gethostbyname` (IPv4 addresses only)
Members of a group by group ID	`vserver services name-service getxxbyyy getgrbygid`
Members of a group by group name	`vserver services name-service getxxbyyy getgrbyname`

List of groups a user belongs to	`vserver services name-service getxxbyyy getgrlist`
Host name of an IP address	`vserver services name-service getxxbyyy getnameinfo` `vserver services name-service getxxbyyy gethostbyaddr` (IPv4 addresses only)
User information by user name	`vserver services name-service getxxbyyy getpwbyname` You can test name resolution of RBAC users by specifying the `-use-rbac` parameter as `true`.
User information by user ID	`vserver services name-service getxxbyyy getpwbyuid` You can test name resolution of RBAC users by specifying the `-use-rbac` parameter as `true`.
Netgroup membership of a client	`vserver services name-service getxxbyyy netgrp`
Netgroup membership of a client using netgroup-by-host search	`vserver services name-service getxxbyyy netgrpbyhost`

The following example shows a DNS lookup test for the SVM vs1 by attempting to obtain the IP address for the host acast1.eng.example.com:

```
cluster1::*> vserver services name-service getxxbyyy getaddrinfo -vserver
vs1 -hostname acast1.eng.example.com -address-family all -show-source true
Source used for lookup: DNS
Host name: acast1.eng.example.com
Canonical Name: acast1.eng.example.com
IPv4: 10.72.8.29
```

The following example shows a NIS lookup test for the SVM vs1 by attempting to retrieve user information for a user with the UID 501768:

```
cluster1::*> vserver services name-service getxxbyyy getpwbyuid -vserver
vs1 -userID 501768 -show-source true
Source used for lookup: NIS
pw_name: jsmith
pw_passwd: $1$y8rA4XX7$/DDOXAvc2PC/IsNFozfIN0
pw_uid: 501768
pw_gid: 501768
pw_gecos:
pw_dir: /home/jsmith
pw_shell: /bin/bash
```

The following example shows an LDAP lookup test for the SVM vs1 by attempting to retrieve user information for a user with the name ldap1:

```
cluster1::*> vserver services name-service getxxbyyy getpwbyname -vserver
vs1 -username ldap1 -use-rbac false -show-source true
Source used for lookup: LDAP
pw_name: ldap1
pw_passwd: {crypt}JSPM6yc/ilIX6
pw_uid: 10001
pw_gid: 3333
pw_gecos: ldap1 user
pw_dir: /u/ldap1
pw_shell: /bin/csh
```

The following example shows a netgroup lookup test for the SVM vs1 by attempting to find out whether the client dnshost0 is a member of the netgroup lnetgroup136:

```
cluster1::*> vserver services name-service getxxbyyy netgrp -vserver vs1
-netgroup lnetgroup136 -client dnshost0 -show-source true
Source used for lookup: LDAP
dnshost0 is a member of lnetgroup136
```

1. Analyze the results of the test you performed and take the necessary action.

If the…	Check the…
Host name or IP address lookup failed or yielded incorrect results	DNS configuration
Lookup queried an incorrect source	Name service switch configuration

User or group lookup failed or yielded incorrect results	Name service switch configuration Source configuration (local files, NIS domain, LDAP client) Network configuration (for example, LIFs and routes)
Host name lookup failed or timed out, and the DNS server does not resolve DNS short names (for example, host1)	DNS configuration for top-level domain (TLD) queries. You can disable TLD queries using the `-is-tld -query-enabled false` option to the `vserver services name-service dns modify` command.

Related information

NetApp Technical Report 4668: Name Services Best Practices Guide

Verify name service connections

Beginning with ONTAP 9.2, you can check DNS and LDAP name servers to verify that they are connected to ONTAP. These commands are available at the admin privilege level.

About this task

You can check for a valid DNS or LDAP name service configuration on an as-needed basis using the name service configuration checker. This validation check can be initiated at the command line or in System Manager.

For DNS configurations, all servers are tested and need to be working for the configuration to be considered valid. For LDAP configurations, as long as any server is up, the configuration is valid. The name service commands apply the configuration checker unless the `skip-config-validation` field is true (the default is false).

Step

1. Use the appropriate command to check a name service configuration. The UI displays the status of the configured servers.

To check…	Use this command…
DNS configuration status	`vserver services name-service dns check`
LDAP configuration status	`vserver services name-service ldap check`

```
cluster1::> vserver services name-service dns check -vserver vs0

Vserver         Name Server     Status  Status Details
--------------  --------------- ------- -------------------------
 vs0            10.11.12.13     up      Response time (msec): 55
 vs0            10.11.12.14     up      Response time (msec): 70
 vs0            10.11.12.15     down    Connection refused.
+-------------------------------------------------------------+
```

```
cluster1::> vserver services name-service ldap check -vserver vs0

| Vserver: vs0                                                |
| Client Configuration Name: c1                               |
| LDAP Status: up                                             |
| LDAP Status Details: Successfully connected to LDAP server
"10.11.12.13".                                              |
```

Configuration validation is successful if at least one of the configured servers (name-servers/ldap-servers) is reachable and providing the service. A warning is shown if some of the servers are not reachable.

Commands for managing name service switch entries

You can manage name service switch entries by creating, displaying, modifying, and deleting them.

If you want to…	Use this command…
Create a name service switch entry	`vserver services name-service ns-switch create`
Display name service switch entries	`vserver services name-service ns-switch show`
Modify a name service switch entry	`vserver services name-service ns-switch modify`
Delete a name service switch entry	`vserver services name-service ns-switch delete`

See the man page for each command for more information.

Related information

NetApp Technical Report 4668: Name Services Best Practices Guide

Commands for managing name service cache

You can manage name service cache by modifying the time to live (TTL) value. The TTL value determines how long name service information is persistent in cache.

If you want to modify the TTL value for…	Use this command…
Unix users	`vserver services name-service cache unix-user settings`
Unix groups	`vserver services name-service cache unix-group settings`
Unix netgroups	`vserver services name-service cache netgroups settings`
Hosts	`vserver services name-service cache hosts settings`
Group membership	`vserver services name-service cache group-membership settings`

Related information

ONTAP 9 Commands

Commands for managing name mappings

There are specific ONTAP commands for managing name mappings.

If you want to…	Use this command…
Create a name mapping	`vserver name-mapping create`
Insert a name mapping at a specific position	`vserver name-mapping insert`
Display name mappings	`vserver name-mapping show`
Exchange the position of two name mappings NOTE: A swap is not allowed when name-mapping is configured with an ip-qualifier entry.	`vserver name-mapping swap`
Modify a name mapping	`vserver name-mapping modify`
Delete a name mapping	`vserver name-mapping delete`
Validate the correct name mapping	`vserver security file-directory show-effective-permissions -vserver vs1 -win-user-name user1 -path / -share-name sh1`

See the man page for each command for more information.

Commands for managing local UNIX users

There are specific ONTAP commands for managing local UNIX users.

If you want to…	Use this command…
Create a local UNIX user	`vserver services name-service unix-user create`
Load local UNIX users from a URI	`vserver services name-service unix-user load-from-uri`
Display local UNIX users	`vserver services name-service unix-user show`
Modify a local UNIX user	`vserver services name-service unix-user modify`
Delete a local UNIX user	`vserver services name-service unix-user delete`

See the man page for each command for more information.

Commands for managing local UNIX groups

There are specific ONTAP commands for managing local UNIX groups.

If you want to…	Use this command…
Create a local UNIX group	`vserver services name-service unix-group create`
Add a user to a local UNIX group	`vserver services name-service unix-group adduser`
Load local UNIX groups from a URI	`vserver services name-service unix-group load-from-uri`
Display local UNIX groups	`vserver services name-service unix-group show`
Modify a local UNIX group	`vserver services name-service unix-group modify`
Delete a user from a local UNIX group	`vserver services name-service unix-group deluser`
Delete a local UNIX group	`vserver services name-service unix-group delete`

See the man page for each command for more information.

Limits for local UNIX users, groups, and group members

ONTAP introduced limits for the maximum number of UNIX users and groups in the cluster, and commands to manage these limits. These limits can help avoid performance issues by preventing administrators from creating too many local UNIX users and groups in the cluster.

There is a limit for the combined number of local UNIX user groups and group members. There is a separate limit for local UNIX users. The limits are cluster-wide. Each of these new limits is set to a default value that you can modify up to a preassigned hard limit.

Database	Default limit	Hard limit
Local UNIX users	32,768	65,536
Local UNIX groups and group members	32,768	65,536

Manage limits for local UNIX users and groups

There are specific ONTAP commands for managing limits for local UNIX users and groups. Cluster administrators can use these commands to troubleshoot performance issues in the cluster believed to be related to excessive numbers of local UNIX users and groups.

About this task

These commands are available to the cluster administrator at the advanced privilege level.

Step

1. Perform one of the following actions:

If you want to…	Use the command…
Display information about local UNIX user limits	`vserver services unix-user max-limit show`
Display information about local UNIX group limits	`vserver services unix-group max-limit show`
Modify local UNIX user limits	`vserver services unix-user max-limit modify`
Modify local UNIX group limits	`vserver services unix-group max-limit modify`

See the man page for each command for more information.

Commands for managing local netgroups

You can manage local netgroups by loading them from a URI, verifying their status across nodes, displaying them, and deleting them.

If you want to…	Use the command…
Load netgroups from a URI	`vserver services name-service netgroup load`
Verify the status of netgroups across nodes	`vserver services name-service netgroup status` Available at the advanced privilege level and higher.
Display local netgroups	`vserver services name-service netgroup file show`
Delete a local netgroup	`vserver services name-service netgroup file delete`

See the man page for each command for more information.

Commands for managing NIS domain configurations

There are specific ONTAP commands for managing NIS domain configurations.

If you want to…	Use this command…
Create a NIS domain configuration	`vserver services name-service nis-domain create`
Display NIS domain configurations	`vserver services name-service nis-domain show`
Display binding status of a NIS domain configuration	`vserver services name-service nis-domain show-bound`
Display NIS statistics	`vserver services name-service nis-domain show-statistics` Available at the advanced privilege level and higher.
Clear NIS statistics	`vserver services name-service nis-domain clear-statistics` Available at the advanced privilege level and higher.
Modify a NIS domain configuration	`vserver services name-service nis-domain modify`
Delete a NIS domain configuration	`vserver services name-service nis-domain delete`
Enable caching for netgroup-by-host searches	`vserver services name-service nis-domain netgroup-database config modify` Available at the advanced privilege level and higher.

See the man page for each command for more information.

Commands for managing LDAP client configurations

There are specific ONTAP commands for managing LDAP client configurations.

SVM administrators cannot modify or delete LDAP client configurations that were created by cluster administrators.

If you want to…	Use this command…
Create an LDAP client configuration	`vserver services name-service ldap client create`
Display LDAP client configurations	`vserver services name-service ldap client show`
Modify an LDAP client configuration	`vserver services name-service ldap client modify`
Change the LDAP client BIND password	`vserver services name-service ldap client modify-bind-password`
Delete an LDAP client configuration	`vserver services name-service ldap client delete`

See the man page for each command for more information.

Commands for managing LDAP configurations

There are specific ONTAP commands for managing LDAP configurations.

If you want to…	Use this command…
Create an LDAP configuration	`vserver services name-service ldap create`
Display LDAP configurations	`vserver services name-service ldap show`
Modify an LDAP configuration	`vserver services name-service ldap modify`
Delete an LDAP configuration	`vserver services name-service ldap delete`

See the man page for each command for more information.

Commands for managing LDAP client schema templates

There are specific ONTAP commands for managing LDAP client schema templates.

SVM administrators cannot modify or delete LDAP client schemas that were created by cluster administrators.

If you want to…	Use this command…
Copy an existing LDAP schema template	`vserver services name-service ldap client schema copy` Available at the advanced privilege level and higher.
Display LDAP schema templates	`vserver services name-service ldap client schema show`
Modify an LDAP schema template	`vserver services name-service ldap client schema modify` Available at the advanced privilege level and higher.
Delete an LDAP schema template	`vserver services name-service ldap client schema delete` Available at the advanced privilege level and higher.

See the man page for each command for more information.

Commands for managing NFS Kerberos interface configurations

There are specific ONTAP commands for managing NFS Kerberos interface configurations.

If you want to…	Use this command…
Enable NFS Kerberos on a LIF	`vserver nfs kerberos interface enable`
Display NFS Kerberos interface configurations	`vserver nfs kerberos interface show`
Modify an NFS Kerberos interface configuration	`vserver nfs kerberos interface modify`
Disable NFS Kerberos on a LIF	`vserver nfs kerberos interface disable`

See the man page for each command for more information.

Commands for managing NFS Kerberos realm configurations

There are specific ONTAP commands for managing NFS Kerberos realm configurations.

If you want to…	Use this command…
Create an NFS Kerberos realm configuration	`vserver nfs kerberos realm create`
Display NFS Kerberos realm configurations	`vserver nfs kerberos realm show`

If you want to…	Use this command…
Modify an NFS Kerberos realm configuration	`vserver nfs kerberos realm modify`
Delete an NFS Kerberos realm configuration	`vserver nfs kerberos realm delete`

See the man page for each command for more information.

Commands for managing export policies

There are specific ONTAP commands for managing export policies.

If you want to…	Use this command…
Display information about export policies	`vserver export-policy show`
Rename an export policy	`vserver export-policy rename`
Copy an export policy	`vserver export-policy copy`
Delete an export policy	`vserver export-policy delete`

See the man page for each command for more information.

Commands for managing export rules

There are specific ONTAP commands for managing export rules.

If you want to…	Use this command…
Create an export rule	`vserver export-policy rule create`
Display information about export rules	`vserver export-policy rule show`
Modify an export rule	`vserver export-policy rule modify`
Delete an export rule	`vserver export-policy rule delete`

If you have configured multiple identical export rules matching different clients, be sure to keep them in sync when managing export rules.

See the man page for each command for more information.

Configure the NFS credential cache

Reasons for modifying the NFS credential cache time-to-live

ONTAP uses a credential cache to store information needed for user authentication for NFS export access to provide faster access and improve performance. You can configure how long information is stored in the credential cache to customize it for your environment.

There are several scenarios when modifying the NFS credential cache time-to-live (TTL) can help resolve issues. You should understand what these scenarios are as well as the consequences of making these modifications.

Reasons

Consider changing the default TTL under the following circumstances:

Issue	Remedial action
The name servers in your environment are experiencing performance degradation due to a high load of requests from ONTAP.	Increase the TTL for cached positive and negative credentials to reduce the number of requests from ONTAP to name servers.
The name server administrator made changes to allow access to NFS users that were previously denied.	Decrease the TTL for cached negative credentials to reduce the time NFS users have to wait for ONTAP to request fresh credentials from external name servers so they can get access.
The name server administrator made changes to deny access to NFS users that were previously allowed.	Reduce the TTL for cached positive credentials to reduce the time before ONTAP requests fresh credentials from external name servers so the NFS users are now denied access.

Consequences

You can modify the length of time individually for caching positive and negative credentials. However, you should be aware of both the advantages and disadvantages of doing so.

If you…	The advantage is…	The disadvantage is…
Increase the positive credential cache time	ONTAP sends requests for credentials to name servers less frequently, reducing the load on name servers.	It takes longer to deny access to NFS users that previously were allowed access but are not anymore.
Decrease the positive credential cache time	It takes less time to deny access to NFS users that previously were allowed access but are not anymore.	ONTAP sends requests for credentials to name servers more frequently, increasing the load on name servers.

If you…	The advantage is…	The disadvantage is…
Increase the negative credential cache time	ONTAP sends requests for credentials to name servers less frequently, reducing the load on name servers.	It takes longer to grant access to NFS users that previously were not allowed access but are now.
Decrease the negative credential cache time	It takes less time to grant access to NFS users that previously were not allowed access but are now.	ONTAP sends requests for credentials to name servers more frequently, increasing the load on name servers.

Configure the time-to-live for cached NFS user credentials

You can configure the length of time that ONTAP stores credentials for NFS users in its internal cache (time-to-live, or TTL) by modifying the NFS server of the storage virtual machine (SVM). This enables you to alleviate certain issues related to high load on name servers or changes in credentials affecting NFS user access.

About this task

These parameters are available at the advanced privilege level.

Steps

1. Set the privilege level to advanced:

   ```
   set -privilege advanced
   ```

2. Perform the desired action:

If you want to modify the TTL for cached…	Use the command…
Positive credentials	`vserver nfs modify -vserver vserver_name -cached -cred-positive-ttl time_to_live` The TTL is measured in milliseconds. The default is 24 hours (86,400,000 milliseconds). The allowed range for this value is 1 minute (60000 milliseconds) through 7 days (604,800,000 milliseconds).
Negative credentials	`vserver nfs modify -vserver vserver_name -cached -cred-negative-ttl time_to_live` The TTL is measured in milliseconds. The default is 2 hours (7,200,000 milliseconds). The allowed range for this value is 1 minute (60000 milliseconds) through 7 days (604,800,000 milliseconds).

3. Return to the admin privilege level:

```
set -privilege admin
```

Manage export policy caches

Flush export policy caches

ONTAP uses several export policy caches to store information related to export policies for faster access. Flushing export policy caches manually (`vserver export-policy cache flush`) removes potentially outdated information and forces ONTAP to retrieve current information from the appropriate external resources. This can help resolve a variety of issues related to client access to NFS exports.

About this task

Export policy cache information might be outdated due to the following reasons:

- A recent change to export policy rules
- A recent change to host name records in name servers
- A recent change to netgroup entries in name servers
- Recovering from a network outage that prevented netgroups from being fully loaded

Steps

1. If you do not have name service cache enabled, perform one of the following actions in advance privilege mode:

If you want to flush…	Enter the command…
All export policy caches (except for showmount)	`vserver export-policy cache flush -vserver vserver_name`
The export policy rules access cache	`vserver export-policy cache flush -vserver vserver_name -cache access` You can include the optional `-node` parameter to specify the node on which you want to flush the access cache.
The host name cache	`vserver export-policy cache flush -vserver vserver_name -cache host`
The netgroup cache	`vserver export-policy cache flush -vserver vserver_name -cache netgroup` Processing of netgroups is resource intensive. You should only flush the netgroup cache if you are trying to resolve a client access issue that is caused by a stale netgroup.
The showmount cache	`vserver export-policy cache flush -vserver vserver_name -cache showmount`

2. If name service cache is enabled, perform one of the following actions:

If you want to flush…	Enter the command…
The export policy rules access cache	`vserver export-policy cache flush -vserver vserver_name -cache access` You can include the optional `-node` parameter to specify the node on which you want to flush the access cache.
The host name cache	`vserver services name-service cache hosts forward-lookup delete-all`
The netgroup cache	`vserver services name-service cache netgroups ip-to-netgroup delete-all` `vserver services name-service cache netgroups members delete-all` Processing of netgroups is resource intensive. You should only flush the netgroup cache if you are trying to resolve a client access issue that is caused by a stale netgroup.
The showmount cache	`vserver export-policy cache flush -vserver vserver_name -cache showmount`

Display the export policy netgroup queue and cache

ONTAP uses the netgroup queue when importing and resolving netgroups and it uses the netgroup cache to store the resulting information. When troubleshooting export policy netgroup related issues, you can use the `vserver export-policy netgroup queue show` and `vserver export-policy netgroup cache show` commands to display the status of the netgroup queue and the contents of the netgroup cache.

Step

1. Perform one of the following actions:

To display the export policy netgroup…	Enter the command…
Queue	`vserver export-policy netgroup queue show`
Cache	`vserver export-policy netgroup cache show -vserver vserver_name`

See the man page for each command for more information.

Check whether a client IP address is a member of a netgroup

When troubleshooting NFS client access issues related to netgroups, you can use the `vserver export-policy netgroup check-membership` command to help determine whether a client IP is a member of a certain netgroup.

About this task

Checking netgroup membership enables you to determine whether ONTAP is aware that a client is or is not member of a netgroup. It also lets you know whether the ONTAP netgroup cache is in a transient state while refreshing netgroup information. This information can help you understand why a client might be unexpectedly granted or denied access.

Step

1. Check the netgroup membership of a client IP address: `vserver export-policy netgroup check-membership -vserver vserver_name -netgroup netgroup_name -client-ip client_ip`

   The command can return the following results:

   - The client is a member of the netgroup.

     This was confirmed through a reverse lookup scan or a netgroup-by-host search.

   - The client is a member of the netgroup.

     It was found in the ONTAP netgroup cache.

   - The client is not a member of the netgroup.
   - The membership of the client cannot yet be determined because ONTAP is currently refreshing the netgroup cache.

     Until this is done, membership cannot be explicitly ruled in or out. Use the `vserver export-policy netgroup queue show` command to monitor the loading of the netgroup and retry the check after it is finished.

Example

The following example checks whether a client with the IP address 172.17.16.72 is a member of the netgroup mercury on the SVM vs1:

```
 cluster1::> vserver export-policy netgroup check-membership -vserver vs1
-netgroup  mercury -client-ip 172.17.16.72
```

Optimize access cache performance

You can configure several parameters to optimize the access cache and find the right balance between performance and how current the information stored in the access cache is.

About this task

When you configure the access cache refresh periods, keep the following in mind:

- Higher values mean entries stay longer in the access cache.

  The advantage is better performance because ONTAP spends less resources on refreshing access cache entries. The disadvantage is that if export policy rules change and access cache entries become stale as a result, it takes longer to update them. As a result, clients that should get access might get denied, and clients that should get denied might get access.

- Lower values mean ONTAP refreshes access cache entries more often.

  The advantage is that entries are more current and clients are more likely to be correctly granted or denied access. The disadvantage is a decrease in performance because ONTAP spends more resources refreshing access cache entries.

Steps

1. Set the privilege level to advanced:

   ```
   set -privilege advanced
   ```

2. Perform the desired action:

To modify the…	Enter…
Refresh period for positive entries	`vserver export-policy access-cache config modify-all-vservers -refresh -period-positive timeout_value`
Refresh period for negative entries	`vserver export-policy access-cache config modify-all-vservers -refresh -period-negative timeout_value`
Timeout period for old entries	`vserver export-policy access-cache config modify-all-vservers -harvest -timeout timeout_value`

3. Verify the new parameter settings:

   ```
   vserver export-policy access-cache config show-all-vservers
   ```

4. Return to the admin privilege level:

   ```
   set -privilege admin
   ```

Manage file locks

About file locking between protocols

File locking is a method used by client applications to prevent a user from accessing a file previously opened by another user. How ONTAP locks files depends on the protocol of the client.

If the client is an NFS client, locks are advisory; if the client is an SMB client, locks are mandatory.

Because of differences between the NFS and SMB file locks, an NFS client might fail to access a file previously opened by an SMB application.

The following occurs when an NFS client attempts to access a file locked by an SMB application:

- In mixed or NTFS volumes, file manipulation operations such as `rm`, `rmdir`, and `mv` can cause the NFS application to fail.
- NFS read and write operations are denied by SMB deny-read and deny-write open modes, respectively.
- NFS write operations fail when the written range of the file is locked with an exclusive SMB bytelock.

In UNIX security-style volumes, NFS unlink and rename operations ignore SMB lock state and allow access to the file. All other NFS operations on UNIX security-style volumes honor SMB lock state.

How ONTAP treats read-only bits

The read-only bit is set on a file-by-file basis to reflect whether a file is writable (disabled) or read-only (enabled).

SMB clients that use Windows can set a per-file read-only bit. NFS clients do not set a per-file read-only bit because NFS clients do not have any protocol operations that use a per-file read-only bit.

ONTAP can set a read-only bit on a file when an SMB client that uses Windows creates that file. ONTAP can also set a read-only bit when a file is shared between NFS clients and SMB clients. Some software, when used by NFS clients and SMB clients, requires the read-only bit to be enabled.

For ONTAP to keep the appropriate read and write permissions on a file shared between NFS clients and SMB clients, it treats the read-only bit according to the following rules:

- NFS treats any file with the read-only bit enabled as if it has no write permission bits enabled.
- If an NFS client disables all write permission bits and at least one of those bits had previously been enabled, ONTAP enables the read-only bit for that file.
- If an NFS client enables any write permission bit, ONTAP disables the read-only bit for that file.
- If the read-only bit for a file is enabled and an NFS client attempts to discover permissions for the file, the permission bits for the file are not sent to the NFS client; instead, ONTAP sends the permission bits to the NFS client with the write permission bits masked.
- If the read-only bit for a file is enabled and an SMB client disables the read-only bit, ONTAP enables the owner's write permission bit for the file.
- Files with the read-only bit enabled are writable only by root.

Changes to file permissions take effect immediately on SMB clients, but might not take effect immediately on NFS clients if the NFS client enables attribute caching.

How ONTAP differs from Windows on handling locks on share path components

Unlike Windows, ONTAP does not lock each component of the path to an open file while the file is open. This behavior also affects SMB share paths.

Because ONTAP does not lock each component of the path, it is possible to rename a path component above

the open file or share, which can cause problems for certain applications, or can cause the share path in the SMB configuration to be invalid. This can cause the share to be inaccessible.

To avoid issues caused by renaming path components, you can apply Windows Access Control List (ACL) security settings that prevent users or applications from renaming critical directories.

Learn more about How to prevent directories from being renamed while clients are accessing them.

Display information about locks

You can display information about the current file locks, including what types of locks are held and what the lock state is, details about byte-range locks, sharelock modes, delegation locks, and opportunistic locks, and whether locks are opened with durable or persistent handles.

About this task

The client IP address cannot be displayed for locks established through NFSv4 or NFSv4.1.

By default, the command displays information about all locks. You can use command parameters to display information about locks for a specific storage virtual machine (SVM) or to filter the command's output by other criteria.

The `vserver locks show` command displays information about four types of locks:

- Byte-range locks, which lock only a portion of a file.
- Share locks, which lock open files.
- Opportunistic locks, which control client-side caching over SMB.
- Delegations, which control client-side caching over NFSv4.x.

By specifying optional parameters, you can determine important information about each lock type. See the man page for the command for more information.

Step

1. Display information about locks by using the `vserver locks show` command.

Examples

The following example displays summary information for an NFSv4 lock on a file with the path `/vol1/file1`. The sharelock access mode is write-deny_none, and the lock was granted with write delegation:

```
cluster1::> vserver locks show

Vserver: vs0
Volume  Object Path               LIF         Protocol  Lock Type   Client
------- ------------------------- ----------- --------- -----------
-------
vol1    /vol1/file1               lif1        nfsv4     share-level -
              Sharelock Mode: write-deny_none
                                                        delegation  -
              Delegation Type: write
```

The following example displays detailed oplock and sharelock information about the SMB lock on a file with the path `/data2/data2_2/intro.pptx`. A durable handle is granted on the file with a share lock access mode of write-deny_none to a client with an IP address of 10.3.1.3. A lease oplock is granted with a batch oplock level:

```
cluster1::> vserver locks show -instance -path /data2/data2_2/intro.pptx

                  Vserver: vs1
                   Volume: data2_2
         Logical Interface: lif2
               Object Path: /data2/data2_2/intro.pptx
                 Lock UUID: 553cf484-7030-4998-88d3-1125adbba0b7
             Lock Protocol: cifs
                 Lock Type: share-level
   Node Holding Lock State: node3
                Lock State: granted
  Bytelock Starting Offset: -
    Number of Bytes Locked: -
     Bytelock is Mandatory: -
     Bytelock is Exclusive: -
     Bytelock is Superlock: -
          Bytelock is Soft: -
              Oplock Level: -
   Shared Lock Access Mode: write-deny_none
       Shared Lock is Soft: false
           Delegation Type: -
            Client Address: 10.3.1.3
              SMB Open Type: durable
         SMB Connect State: connected
SMB Expiration Time (Secs): -
         SMB Open Group ID:
78a90c59d45ae211998100059a3c7a00a007f70da0f8ffffcd445b0300000000

                  Vserver: vs1
```

```
                    Volume: data2_2
          Logical Interface: lif2
                Object Path: /data2/data2_2/test.pptx
                  Lock UUID: 302fd7b1-f7bf-47ae-9981-f0dcb6a224f9
              Lock Protocol: cifs
                  Lock Type: op-lock
   Node Holding Lock State: node3
                Lock State: granted
  Bytelock Starting Offset: -
    Number of Bytes Locked: -
     Bytelock is Mandatory: -
     Bytelock is Exclusive: -
     Bytelock is Superlock: -
          Bytelock is Soft: -
              Oplock Level: batch
   Shared Lock Access Mode: -
       Shared Lock is Soft: -
           Delegation Type: -
            Client Address: 10.3.1.3
             SMB Open Type: -
         SMB Connect State: connected
SMB Expiration Time (Secs): -
         SMB Open Group ID:
78a90c59d45ae211998100059a3c7a00a007f70da0f8ffffcd445b0300000000
```

Breaking locks

When file locks are preventing client access to files, you can display information about currently held locks, and then break specific locks. Examples of scenarios in which you might need to break locks include debugging applications.

About this task

The `vserver locks break` command is available only at the advanced privilege level and higher. The man page for the command contains detailed information.

Steps

1. To find the information you need to break a lock, use the `vserver locks show` command.

   The man page for the command contains detailed information.

2. Set the privilege level to advanced:

   `set -privilege advanced`

3. Perform one of the following actions:

If you want to break a lock by specifying…	Enter the command…

The SVM name, volume name, LIF name, and file path	`vserver locks break -vserver vserver_name -volume volume_name -path path -lif lif`
The lock ID	`vserver locks break -lockid UUID`

4. Return to the admin privilege level:

   `set -privilege admin`

How FPolicy first-read and first-write filters work with NFS

NFS clients experience high response time during high traffic of read/write requests when the FPolicy is enabled using an external FPolicy server with read/write operations as monitored events. For NFS clients, the use of first-read and first-write filters in the FPolicy reduces the number of FPolicy notifications and improves performance.

In NFS, the client does I/O on a file by fetching its handle. This handle might remain valid across reboots of the server and the client. Therefore, the client is free to cache the handle and send requests on it without retrieving handles again. In a regular session, lots of reads/write requests are sent to the file server. If notifications are generated for all these requests, it might result in the following issues:

- A larger load due to additional notification processing, and higher response time.
- A large number of notifications being sent to the FPolicy server even though the server unaffected by all of the notifications.

After receiving the first read/write request from a client for a particular file, a cache entry is created and the read/write count is incremented. This request is marked as the first-read/write operation, and an FPolicy event is generated. Before you plan and create your FPolicy filters for an NFS client, you should understand the basics of how FPolicy filters work.

- First-read: Filters the client read requests for first-read.

  When this filter is used for NFS events, the `-file-session-io-grouping-count` and `-file -session-io-grouping-duration` settings determine the first-read request for which FPolicy is processed.

- First-write: Filters the client write requests for first-write.

  When this filter is used for NFS events, the `-file-session-io-grouping-count` and `-file -session-io-grouping-duration` settings determine the first-write request for which FPolicy processed.

The following options are added in NFS servers database.

```
file-session-io-grouping-count: Number of I/O Ops on a File to Be Clubbed
and Considered as One Session
for Event Generation
file-session-io-grouping-duration: Duration for Which I/O Ops on a File to
Be Clubbed and Considered as
One Session for Event Generation
```

Modify the NFSv4.1 server implementation ID

The NFSv4.1 protocol includes a server implementation ID that documents the server domain, name, and date. You can modify the server implementation ID default values. Changing the default values can be useful, for example, when gathering usage statistics or troubleshooting interoperability issues. For more information, see RFC 5661.

About this task

The default values for the three options are as follows:

Option	Option name	Default value
NFSv4.1 Implementation ID Domain	`-v4.1-implementation -domain`	netapp.com
NFSv4.1 Implementation ID Name	`-v4.1-implementation-name`	Cluster version name
NFSv4.1 Implementation ID Date	`-v4.1-implementation-date`	Cluster version date

Steps

1. Set the privilege level to advanced:

   ```
   set -privilege advanced
   ```

2. Perform one of the following actions:

If you want to modify the NFSv4.1 implementation ID…	Enter the command…
Domain	`vserver nfs modify -v4.1 -implementation-domain domain`
Name	`vserver nfs modify -v4.1 -implementation-name name`
Date	`vserver nfs modify -v4.1 -implementation-date date`

3. Return to the admin privilege level:

```
set -privilege admin
```

Manage NFSv4 ACLs

Benefits of enabling NFSv4 ACLs

There are many benefits to enabling NFSv4 ACLs.

The benefits of enabling NFSv4 ACLs include the following:

- Finer-grained control of user access for files and directories
- Better NFS security
- Improved interoperability with CIFS
- Removal of the NFS limitation of 16 groups per user

How NFSv4 ACLs work

A client using NFSv4 ACLs can set and view ACLs on files and directories on the system. When a new file or subdirectory is created in a directory that has an ACL, the new file or subdirectory inherits all ACL Entries (ACEs) in the ACL that have been tagged with the appropriate inheritance flags.

When a file or directory is created as the result of an NFSv4 request, the ACL on the resulting file or directory depends on whether the file creation request includes an ACL or only standard UNIX file access permissions, and whether the parent directory has an ACL:

- If the request includes an ACL, that ACL is used.
- If the request includes only standard UNIX file access permissions but the parent directory has an ACL, the ACEs in the parent directory's ACL are inherited by the new file or directory as long as the ACEs have been tagged with the appropriate inheritance flags.

  A parent ACL is inherited even if `-v4.0-acl` is set to `off`.

- If the request includes only standard UNIX file access permissions and the parent directory does not have an ACL, the client file mode is used to set standard UNIX file access permissions.
- If the request includes only standard UNIX file access permissions and the parent directory has a non-inheritable ACL, the new object is created only with mode bits.

If the `-chown-mode` parameter has been set to `restricted` with commands in the `vserver nfs` or `vserver export-policy rule` families, file ownership can be changed by the superuser only, even if the on-disk permissions set with NFSv4 ACLs allow a non-root user to change the file ownership. For more information, see the relevant man pages.

Enable or disable modification of NFSv4 ACLs

When ONTAP receives a `chmod` command for a file or directory with an ACL, by default

the ACL is retained and modified to reflect the mode bit change. You can disable the `-v4-acl-preserve` parameter to change the behavior if you want the ACL to be dropped instead.

About this task

When using unified security style, this parameter also specifies whether NTFS file permissions are preserved or dropped when a client sends a chmod, chgroup, or chown command for a file or directory.

The default for this parameter is enabled.

Steps

1. Set the privilege level to advanced:

   ```
   set -privilege advanced
   ```

2. Perform one of the following actions:

If you want to…	Enter the following command…
Enable retention and modification of existing NFSv4 ACLs (default)	`vserver nfs modify -vserver vserver_name -v4-acl-preserve enabled`
Disable retention and drop NFSv4 ACLs when changing mode bits	`vserver nfs modify -vserver vserver_name -v4-acl-preserve disabled`

3. Return to the admin privilege level:

   ```
   set -privilege admin
   ```

How ONTAP uses NFSv4 ACLs to determine whether it can delete a file

To determine whether it can delete a file, ONTAP uses a combination of the file's DELETE bit, and the containing directory's DELETE_CHILD bit. For more information, see the NFS 4.1 RFC 5661.

Enable or disable NFSv4 ACLs

To enable or disable NFSv4 ACLs, you can modify the `-v4.0-acl` and `-v4.1-acl` options. These options are disabled by default.

About this task

The `-v4.0-acl` or `-v4.1-acl` option controls the setting and viewing of NFSv4 ACLs; it does not control enforcement of these ACLs for access checking.

Step

1. Perform one of the following actions:

If you want to…	Then…

Enable NFSv4.0 ACLs	Enter the following command: `vserver nfs modify -vserver vserver_name -v4.0-acl enabled`
Disable NFSv4.0 ACLs	Enter the following command: `vserver nfs modify -vserver vserver_name -v4.0-acl disabled`
Enable NFSv4.1 ACLs	Enter the following command: `vserver nfs modify -vserver vserver_name -v4.1-acl enabled`
Disable NFSv4.1 ACLs	Enter the following command: `vserver nfs modify -vserver vserver_name -v4.1-acl disabled`

Modify the maximum ACE limit for NFSv4 ACLs

You can modify the maximum number of allowed ACEs for each NFSv4 ACL by modifying the parameter `-v4-acl-max-aces`. By default, the limit is set to 400 ACEs for each ACL. Increasing this limit can help ensure successful migration of data with ACLs containing over 400 ACEs to storage systems running ONTAP.

About this task

Increasing this limit might impact performance for clients accessing files with NFSv4 ACLs.

Steps

1. Set the privilege level to advanced:

   ```
   set -privilege advanced
   ```

2. Modify the maximum ACE limit for NFSv4 ACLs:

   ```
   vserver nfs modify -v4-acl-max-aces max_ace_limit
   ```

   The valid range of

   `max_ace_limit` is `192` to `1024`.

3. Return to the admin privilege level:

   ```
   set -privilege admin
   ```

Manage NFSv4 file delegations

Enable or disable NFSv4 read file delegations

To enable or disable NFSv4 read file delegations, you can modify the `-v4.0-read-delegation`or `-v4.1-read-delegation` option. By enabling read file delegations, you can eliminate much of the message overhead associated with the opening and closing of files.

About this task

By default, read file delegations are disabled.

The disadvantage of enabling read file delegations is that the server and its clients must recover delegations after the server reboots or restarts, a client reboots or restarts, or a network partition occurs.

Step

1. Perform one of the following actions:

If you want to…	Then…
Enable NFSv4 read file delegations	Enter the following command: `vserver nfs modify -vserver vserver_name -v4.0 -read-delegation enabled`
Enable NFSv4.1 read file delegations	Enter the following command: + `vserver nfs modify -vserver vserver_name -v4.1 -read-delegation enabled`
Disable NFSv4 read file delegations	Enter the following command: `vserver nfs modify -vserver vserver_name -v4.0 -read-delegation disabled`
Disable NFSv4.1 read file delegations	Enter the following command: `vserver nfs modify -vserver vserver_name -v4.1 -read-delegation disabled`

Result

The file delegation options take effect as soon as they are changed. There is no need to reboot or restart NFS.

Enable or disable NFSv4 write file delegations

To enable or disable write file delegations, you can modify the `-v4.0-write-delegation`or `-v4.1-write-delegation` option. By enabling write file delegations, you can eliminate much of the message overhead associated with file and record locking

in addition to opening and closing of files.

About this task

By default, write file delegations are disabled.

The disadvantage of enabling write file delegations is that the server and its clients must perform additional tasks to recover delegations after the server reboots or restarts, a client reboots or restarts, or a network partition occurs.

Step

1. Perform one of the following actions:

If you want to…	Then…
Enable NFSv4 write file delegations	Enter the following command: `vserver nfs modify -vserver vserver_name -v4.0 -write-delegation enabled`
Enable NFSv4.1 write file delegations	Enter the following command: `vserver nfs modify -vserver vserver_name -v4.1 -write-delegation enabled`
Disable NFSv4 write file delegations	Enter the following command: `vserver nfs modify -vserver vserver_name -v4.0 -write-delegation disabled`
Disable NFSv4.1 write file delegations	Enter the following command: `vserver nfs modify -vserver vserver_name -v4.1 -write-delegation disabled`

Result

The file delegation options take effect as soon as they are changed. There is no need to reboot or restart NFS.

Configure NFSv4 file and record locking

About NFSv4 file and record locking

For NFSv4 clients, ONTAP supports the NFSv4 file-locking mechanism, maintaining the state of all file locks under a lease-based model.

NetApp Technical Report 3580: NFSv4 Enhancements and Best Practices Guide Data ONTAP Implementation

Specify the NFSv4 locking lease period

To specify the NFSv4 locking lease period (that is, the time period in which ONTAP irrevocably grants a lock to a client), you can modify the `-v4-lease-seconds` option. Shorter lease periods speed up server recovery while longer lease periods are beneficial for servers handling a very large amount of clients.

About this task

By default, this option is set to `30`. The minimum value for this option is `10`. The maximum value for this option is the locking grace period, which you can set with the `locking.lease_seconds` option.

Steps

1. Set the privilege level to advanced:

   ```
   set -privilege advanced
   ```

2. Enter the following command:

   ```
   vserver nfs modify -vserver vserver_name -v4-lease-seconds number_of_seconds
   ```

3. Return to the admin privilege level:

   ```
   set -privilege admin
   ```

Specify the NFSv4 locking grace period

To specify the NFSv4 locking grace period (that is, the time period in which clients attempt to reclaim their locking state from ONTAP during server recovery), you can modify the `-v4-grace-seconds` option.

About this task

By default, this option is set to `45`.

Steps

1. Set the privilege level to advanced:

   ```
   set -privilege advanced
   ```

2. Enter the following command:

   ```
   vserver nfs modify -vserver vserver_name -v4-grace-seconds number_of_seconds
   ```

3. Return to the admin privilege level:

   ```
   set -privilege admin
   ```

How NFSv4 referrals work

When you enable NFSv4 referrals, ONTAP provides “intra-SVM” referrals to NFSv4 clients. Intra-SVM referral is when a cluster node receiving the NFSv4 request refers the NFSv4 client to another logical interface (LIF) on the storage virtual machine (SVM).

The NFSv4 client should access the path that received the referral at the target LIF from that point onward. The original cluster node provides such a referral when it determines that there exists a LIF in the SVM that is resident on the cluster node on which the data volume resides, thereby enabling the clients faster access to the data and avoiding extra cluster communication.

Enable or disable NFSv4 referrals

You can enable NFSv4 referrals on storage virtual machines (SVMs) by enabling the options `-v4-fsid-change` and `-v4.0-referrals`or `-v4.1-referrals`. Enabling NFSV4 referrals can result in faster data access for NFSv4 clients that support this feature.

What you'll need

If you want to enable NFS referrals, you must first disable parallel NFS. You cannot enable both at the same time.

Steps

1. Set the privilege level to advanced:

   ```
   set -privilege advanced
   ```

2. Perform one of the following actions:

If you want to…	Enter the command…
Enable NFSv4 referrals	`vserver nfs modify -vserver vserver_name -v4-fsid -change enabled vserver nfs modify -vserver vserver_name -v4.0-referrals enabled`
Disable NFSv4 referrals	`vserver nfs modify -vserver vserver_name -v4.0 -referrals disabled`
Enable NFSv4.1 referrals	`vserver nfs modify -vserver vserver_name -v4-fsid -change enabled vserver nfs modify -vserver vserver_name -v4.1-referrals enabled`
Disable NFSv4.1 referrals	`vserver nfs modify -vserver vserver_name -v4.1 -referrals disabled`

3. Return to the admin privilege level:

   ```
   set -privilege admin
   ```

Display NFS statistics

You can display NFS statistics for storage virtual machines (SVMs) on the storage system to monitor performance and diagnose issues.

Steps

1. Use the `statistics catalog object show` command to identify the NFS objects from which you can view data.

   ```
   statistics catalog object show -object nfs*
   ```

2. Use the `statistics start` and optional `statistics stop` commands to collect a data sample from one or more objects.
3. Use the `statistics show` command to view the sample data.

Example: Monitoring NFSv3 performance

The following example shows performance data for the NFSv3 protocol.

The following command starts data collection for a new sample:

```
vs1::> statistics start -object nfsv3 -sample-id nfs_sample
```

The following command shows data from the sample by specifying counters that show the number of successful read and write requests versus the total number of read and write requests:

```
vs1::> statistics show -sample-id nfs_sample -counter
read_total|write_total|read_success|write_success

Object: nfsv3
Instance: vs1
Start-time: 2/11/2013 15:38:29
End-time: 2/11/2013 15:38:41
Cluster: cluster1

    Counter                                     Value
    --------------------------- ----------------------------
    read_success                                  40042
    read_total                                    40042
    write_success                               1492052
    write_total                                 1492052
```

Related information

Performance monitoring setup

Display DNS statistics

You can display DNS statistics for storage virtual machines (SVMs) on the storage system to monitor performance and diagnose issues.

Steps

1. Use the `statistics catalog object show` command to identify the DNS objects from which you can view data.

   ```
   statistics catalog object show -object external_service_op*
   ```

2. Use the `statistics start` and `statistics stop` commands to collect a data sample from one or more objects.

3. Use the `statistics show` command to view the sample data.

Monitoring DNS statistics

The following examples show performance data for DNS queries. The following commands start data collection for a new sample:

```
vs1::*> statistics start -object external_service_op -sample-id
dns_sample1
vs1::*> statistics start -object external_service_op_error -sample-id
dns_sample2
```

The following command displays data from the sample by specifying counters that display the number of DNS queries sent versus the number of DNS queries received, failed, or timed out:

```
vs1::*> statistics show -sample-id dns_sample1 -counter
num_requests_sent|num_responses_received|num_successful_responses|num_time
outs|num_request_failures|num_not_found_responses

Object: external_service_op
Instance: vs1:DNS:Query:10.72.219.109
Start-time: 3/8/2016 11:15:21
End-time: 3/8/2016 11:16:52
Elapsed-time: 91s
Scope: vs1

    Counter                                                     Value
    -------------------------------- --------------------------------
    num_not_found_responses                                         0
    num_request_failures                                            0
    num_requests_sent                                               1
    num_responses_received                                          1
    num_successful_responses                                        1
    num_timeouts                                                    0
6 entries were displayed.
```

The following command displays data from the sample by specifying counters that display the number of times a specific error was received for a DNS query on the particular server:

```
vs1::*> statistics show -sample-id dns_sample2 -counter
server_ip_address|error_string|count

Object: external_service_op_error
Instance: vs1:DNS:Query:NXDOMAIN:10.72.219.109
Start-time: 3/8/2016 11:23:21
End-time: 3/8/2016 11:24:25
Elapsed-time: 64s
Scope: vs1

    Counter                                                     Value
    -------------------------------- --------------------------------
    count                                                           1
    error_string                                             NXDOMAIN
    server_ip_address                                   10.72.219.109
3 entries were displayed.
```

Related information

Performance monitoring setup

Display NIS statistics

You can display NIS statistics for storage virtual machines (SVMs) on the storage system to monitor performance and diagnose issues.

Steps

1. Use the `statistics catalog object show` command to identify the NIS objects from which you can view data.

   ```
   statistics catalog object show -object external_service_op*
   ```

2. Use the `statistics start` and `statistics stop` commands to collect a data sample from one or more objects.
3. Use the `statistics show` command to view the sample data.

Monitoring NIS statistics

The following examples display performance data for NIS queries. The following commands start data collection for a new sample:

```
vs1::*> statistics start -object external_service_op -sample-id
nis_sample1
vs1::*> statistics start -object external_service_op_error -sample-id
nis_sample2
```

The following command displays data from the sample by specifying counters that show the number of NIS

queries sent versus the number of NIS queries received, failed, or timed out:

```
vs1::*> statistics show -sample-id nis_sample1 -counter
instance|num_requests_sent|num_responses_received|num_successful_responses
|num_timeouts|num_request_failures|num_not_found_responses

Object: external_service_op
Instance: vs1:NIS:Query:10.227.13.221
Start-time: 3/8/2016 11:27:39
End-time: 3/8/2016 11:27:56
Elapsed-time: 17s
Scope: vs1

    Counter                                                     Value
    -------------------------------- --------------------------------
    num_not_found_responses                                         0
    num_request_failures                                            1
    num_requests_sent                                               2
    num_responses_received                                          1
    num_successful_responses                                        1
    num_timeouts                                                    0
6 entries were displayed.
```

The following command displays data from the sample by specifying counters that show the number of times a specific error was received for a NIS query on the particular server:

```
vs1::*> statistics show -sample-id nis_sample2 -counter
server_ip_address|error_string|count

Object: external_service_op_error
Instance: vs1:NIS:Query:YP_NOTFOUND:10.227.13.221
Start-time: 3/8/2016 11:33:05
End-time: 3/8/2016 11:33:10
Elapsed-time: 5s
Scope: vs1

    Counter                                                     Value
    -------------------------------- --------------------------------
    count                                                           1
    error_string                                            YP_NOTFOUND
    server_ip_address                                     10.227.13.221
3 entries were displayed.
```

Related information

Performance monitoring setup

Support for VMware vStorage over NFS

ONTAP supports certain VMware vStorage APIs for Array Integration (VAAI) features in an NFS environment.

Supported features

The following features are supported:

- Copy offload

  Enables an ESXi host to copy virtual machines or virtual machine disks (VMDKs) directly between the source and destination data store location without involving the host. This conserves ESXi host CPU cycles and network bandwidth. Copy offload preserves space efficiency if the source volume is sparse.

- Space reservation

  Guarantees storage space for a VMDK file by reserving space for it.

Limitations

VMware vStorage over NFS has the following limitations:

- Copy offload operations can fail in the following scenarios:
  - While running wafliron on the source or destination volume because it temporarily takes the volume offline
  - While moving either the source or destination volume
  - While moving either the source or destination LIF
  - While performing takeover or giveback operations
  - While performing switchover or switchback operations
- Server-side copy can fail due to file handle format differences in the following scenario:

  You attempt to copy data from SVMs that have currently or had previously exported qtrees to SVMs that have never had exported qtrees. To work around this limitation, you can export at least one qtree on the destination SVM.

Related information

What VAAI offloaded operations are supported by Data ONTAP?

Enable or disable VMware vStorage over NFS

You can enable or disable support for VMware vStorage over NFS on storage virtual machines (SVMs) by using the `vserver nfs modify` command.

About this task

By default, support for VMware vStorage over NFS is disabled.

Steps

1. Display the current vStorage support status for SVMs:

```
vserver nfs show -vserver vserver_name -instance
```

2. Perform one of the following actions:

If you want to…	Enter the following command…
Enable VMware vStorage support	`vserver nfs modify -vserver vserver_name -vstorage enabled`
Disable VMware vStorage support	`vserver nfs modify -vserver vserver_name -vstorage disabled`

After you finish

You must install the NFS Plug-in for VMware VAAI before you can use this functionality. For more information, see *Installing the NetApp NFS Plug-in for VMware VAAI*.

Related information

NetApp Documentation: NetApp NFS Plug-in for VMware VAAI

Enable or disable rquota support

ONTAP supports the remote quota protocol version 1 (rquota v1). The rquota protocol enables NFS clients to obtain quota information for users from a remote machine. You can enable rquota on storage virtual machines (SVMs) by using the `vserver nfs modify` command.

About this task

By default, rquota is disabled.

Step

1. Perform one of the following actions:

If you want to…	Enter the following command…
Enable rquota support for SVMs	`vserver nfs modify -vserver vserver_name -rquota enable`
Disable rquota support for SVMs	`vserver nfs modify -vserver vserver_name -rquota disable`

For more information about quotas, see Logical storage management.

NFSv3 and NFSv4 performance improvement by modifying the TCP transfer size

You can improve the performance of NFSv3 and NFSv4 clients connecting to storage systems over a high-latency network by modifying the TCP maximum transfer size.

When clients access storage systems over a high-latency network, such as a wide area network (WAN) or metro area network (MAN) with a latency over 10 milliseconds, you might be able to improve the connection performance by modifying the TCP maximum transfer size. Clients accessing storage systems in a low-latency network, such as a local area network (LAN), can expect little to no benefit from modifying these parameters. If the throughput improvement does not outweigh the latency impact, you should not use these parameters.

To determine whether your storage environment would benefit from modifying these parameters, you should first conduct a comprehensive performance evaluation of a poorly performing NFS client. Review whether the low performance is because of excessive round trip latency and small request on the client. Under these conditions, the client and server cannot fully use the available bandwidth because they spend the majority of their duty cycles waiting for small requests and responses to be transmitted over the connection.

By increasing the NFSv3 and NFSv4 request size, the client and server can use the available bandwidth more effectively to move more data per unit time; therefore, increasing the overall efficiency of the connection.

Keep in mind that the configuration between the storage system and the client might vary. The storage system and the client supports maximum size of 1 MB for transfer operations. However, if you configure the storage system to support 1 MB maximum transfer size but the client only supports 64 KB, then the mount transfer size is limited to 64 KB or less.

Before modifying these parameters, you must be aware that it results in additional memory consumption on the storage system for the period of time necessary to assemble and transmit a large response. The more high-latency connections to the storage system, the higher the additional memory consumption. Storage systems with high memory capacity might experience very little effect from this change. Storage systems with low memory capacity might experience noticeable performance degradation.

The successful use of these parameter relies on the ability to retrieve data from multiple nodes of a cluster. The inherent latency of the cluster network might increase the overall latency of the response. Overall latency tends to increase when using these parameters. As a result, latency sensitive workloads might show negative impact.

Modify the NFSv3 and NFSv4 TCP maximum transfer size

You can modify the `-tcp-max-xfer-size` option to configure maximum transfer sizes for all TCP connections using the NFSv3 and NFSv4.x protocols.

About this task

You can modify these options individually for each storage virtual machine (SVM).

Beginning with ONTAP 9, the `v3-tcp-max-read-size` and `v3-tcp-max-write-size` options are obsolete. You must use the `-tcp-max-xfer-size` option instead.

Steps

1. Set the privilege level to advanced:

   ```
   set -privilege advanced
   ```

2. Perform one of the following actions:

If you want to…	Enter the command…
Modify the NFSv3 or NFSv4 TCP maximum transfer size	`vserver nfs modify -vserver vserver_name -tcp-max-xfer-size integer_max_xfer_size`

Option	Range	Default
`-tcp-max-xfer-size`	8192 to 1048576 bytes	65536 bytes

The maximum transfer size that you enter must be a multiple of 4 KB (4096 bytes). Requests that are not properly aligned negatively affect performance.

3. Use the `vserver nfs show -fields tcp-max-xfer-size` command to verify the changes.
4. If any clients use static mounts, unmount and remount for the new parameter size to take effect.

Example

The following command sets the NFSv3 and NFSv4.x TCP maximum transfer size to 1048576 bytes on the SVM named vs1:

```
vs1::> vserver nfs modify -vserver vs1 -tcp-max-xfer-size 1048576
```

Configure the number of group IDs allowed for NFS users

By default, ONTAP supports up to 32 group IDs when handling NFS user credentials using Kerberos (RPCSEC_GSS) authentication. When using AUTH_SYS authentication, the default maximum number of group IDs is 16, as defined in RFC 5531. You can increase the maximum up to 1,024 if you have users who are members of more than the default number of groups.

About this task

If a user has more than the default number of group IDs in their credentials, the remaining group IDs are truncated and the user might receive errors when attempting to access files from the storage system. You should set the maximum number of groups, per SVM, to a number that represents the maximum groups in your environment.

The following table shows the two parameters of the `vserver nfs modify` command that determine the maximum number of group IDs in three sample configurations:

Parameters	Settings	Resulting group IDs limit

`-extended-groups-limit`
 	
`-auth-sys-extended-groups`
	`32`
	
	`disabled`
	
	These are the default settings.
		RPCSEC_GSS: 32
		
		AUTH_SYS: 16

`-extended-groups-limit`
 	
`-auth-sys-extended-groups`
	`256`
	
	`disabled`
		RPCSEC_GSS: 256
		
		AUTH_SYS: 16

`-extended-groups-limit`
 	
`-auth-sys-extended-groups`
	`512`
	
	`enabled`
		RPCSEC_GSS: 512
		
		AUTH_SYS: 512

Some older NFS clients might not be compatible with AUTH_SYS extended groups.

Steps

1. Set the privilege level to advanced:

   ```
   set -privilege advanced
   ```

2. Perform the desired action:

If you want to set the maximum number of allowed auxiliary groups…	Enter the command…
Only for RPCSEC_GSS and leave AUTH_SYS set to the default value of 16	`vserver nfs modify -vserver vserver_name -extended-groups-limit {32-1024} -auth-sys-extended-groups disabled`
For both RPCSEC_GSS and AUTH_SYS	`vserver nfs modify -vserver vserver_name -extended-groups-limit {32-1024} -auth-sys-extended-groups enabled`

3. Verify the `-extended-groups-limit` value and verify whether AUTH_SYS is using extended groups: `vserver nfs show -vserver vserver_name -fields auth-sys-extended-groups,extended-groups-limit`
4. Return to the admin privilege level:

   ```
   set -privilege admin
   ```

Example

The following example enables extended groups for AUTH_SYS authentication and sets the maximum number of extended groups to 512 for both AUTH_SYS and RPCSEC_GSS authentication. These changes are made only for clients who access the SVM named vs1:

```
vs1::> set -privilege advanced
Warning: These advanced commands are potentially dangerous; use
         them only when directed to do so by NetApp personnel.
Do you want to continue? {y|n}: y

vs1::*> vserver nfs modify -vserver vs1 -auth-sys-extended-groups enabled
-extended-groups-limit 512

vs1::*> vserver nfs show -vserver vs1 -fields auth-sys-extended-
groups,extended-groups-limit
vserver auth-sys-extended-groups extended-groups-limit
------- ------------------------ --------------------
vs1     enabled                  512

vs1::*> set -privilege admin
```

Control root user access to NTFS security-style data

You can configure ONTAP to allow NFS clients access to NTFS security-style data and NTFS clients to access NFS security-style data. When using NTFS security style on an NFS data store, you must decide how to treat access by the root user and configure the storage virtual machine (SVM) accordingly.

About this task

When a root user accesses NTFS security-style data, you have two options:

- Map the root user to a Windows user like any other NFS user and manage access according to NTFS ACLs.
- Ignore NTFS ACLs and provide full access to root.

Steps

1. Set the privilege level to advanced:

   ```
   set -privilege advanced
   ```

2. Perform the desired action:

If you want the root user to…	Enter the command…
Be mapped to a Windows user	`vserver nfs modify -vserver vserver_name -ignore -nt-acl-for-root disabled`
Bypass the NT ACL check	`vserver nfs modify -vserver vserver_name -ignore -nt-acl-for-root enabled`

   By default, this parameter is disabled.

If this parameter is enabled but there is no name mapping for the root user, ONTAP uses a default SMB administrator credential for auditing.

3. Return to the admin privilege level:

```
set -privilege admin
```

Supported NFS versions and clients

Overview of supported NFS versions and clients

Before you can use NFS in your network, you need to know which NFS versions and clients ONTAP supports.

This table notes when major and minor NFS protocol versions are supported by default in ONTAP. Support by default does not indicate that this is the earliest version of ONTAP supporting that NFS protocol.

Version	Enabled by default
NFSv3	Yes
NFSv4.0	Yes, beginning with ONTAP 9.9.1
NFSv4.1	Yes, beginning with ONTAP 9.9.1
NFSv4.2	Yes, beginning with ONTAP 9.9.1
pNFS	No

For the latest information about which NFS clients ONTAP supports, see the Interoperability Matrix.

NetApp Interoperability Matrix Tool

NFSv4.0 functionality supported by ONTAP

ONTAP supports all the mandatory functionality in NFSv4.0 except the SPKM3 and LIPKEY security mechanisms.

The following NFSV4 functionality is supported:

- **COMPOUND**

  Allows a client to request multiple file operations in a single remote procedure call (RPC) request.

- **File delegation**

  Allows the server to delegate file control to some types of clients for read and write access.

- **Pseudo-fs**

Used by NFSv4 servers to determine mount points on the storage system. There is no mount protocol in NFSv4.

- **Locking**

  Lease-based. There are no separate Network Lock Manager (NLM) or Network Status Monitor (NSM) protocols in NFSv4.

For more information about the NFSv4.0 protocol, see RFC 3530.

Limitations of ONTAP support for NFSv4

You should be aware of several limitations of ONTAP support for NFSv4.

- The delegation feature is not supported by every client type.
- In ONTAP 9.4 and earlier releases, names with non-ASCII characters on volumes other than UTF8 volumes are rejected by the storage system.

  In ONTAP 9.5 and later releases, volumes created with the utf8mb4 language setting and mounted using NFS v4 are no longer subject to this restriction.

- All file handles are persistent; the server does not give volatile file handles.
- Migration and replication are not supported.
- NFSv4 clients are not supported with read-only load-sharing mirrors.

  ONTAP routes NFSv4 clients to the source of the load-sharing mirror for direct read and write access.

- Named attributes are not supported.
- All recommended attributes are supported, except for the following:

  - `archive`
  - `hidden`
  - `homogeneous`
  - `mimetype`
  - `quota_avail_hard`
  - `quota_avail_soft`
  - `quota_used`
  - `system`
  - `time_backup`

  Although it does not support the `quota*` attributes, ONTAP does support user and group quotas through the RQUOTA side band protocol.

ONTAP support for NFSv4.1

Beginning with ONTAP 9.8, nconnect functionality is available by default when NFSv4.1 is

enabled.

Earlier NFS client implementations use only a single TCP connection with a mount. In ONTAP, a single TCP connection can become a bottleneck with increasing IOPS. However, an nconnect-enabled client can have multiple TCP connections (up to 16) associated with a single NFS mount. Such an NFS client multiplexes file operations onto multiple TCP connections in a round-robin fashion and thus obtains higher throughput from the available network bandwidth. Nconnect is recommended for NFSv3 and NFSv4.1 mounts only.

See your NFS client documentation to confirm whether nconnect is supported in your client version.

NFSv4.1 is enabled by default in ONTAP 9.9.1 and later. In earlier releases, you can enable it by specifying the `-v4.1` option and setting it to `enabled` when creating an NFS server on the storage virtual machine (SVM).

ONTAP does not support NFSv4.1 directory and file level delegations.

ONTAP support for NFSv4.2

Beginning with ONTAP 9.8, the NFSv4.2 protocol is supported to allow access for NFSv4.2 clients.

NFSv4.2 is enabled by default in ONTAP 9.9.1 and later. In ONTAP 9.8, you can enable v4.2 by specifying the `-v4.1` option and setting it to `enabled` when creating an NFS server on the storage virtual machine (SVM). Enabling NFSv4.1 also enables clients to use the NFSv4.1 features while mounted as v4.2.

The following NFSv4.2 optional features are supported:

- Beginning with ONTAP 9.9.1, Mandatory Access Control (MAC) labelled NFS is supported when NFSv4.2 is enabled.
- Additional NFSv4.2 optional features will be added in a later ONTAP release.

Enable NFS v4.2 security labels

Beginning with ONTAP 9.9.1, NFS security labels can be enabled. They are disabled by default.

With NFS v4.2 security labels, ONTAP NFS servers are Mandatory Access Control (MAC) aware, storing and retrieving sec_label attributes sent by clients.

For more information, see RFC7240

> NFS v4.2 security labels are not currently supported for NDMP dump operations. If security labels are encountered on files or directories, the dump fails.

Steps

1. Change the privilege setting to advanced:

   ```
   set -privilege advanced
   ```

2. Enable security labels:

   ```
   vserver nfs modify -vserver svm_name -v4.2-seclabel enabled
   ```

ONTAP support for parallel NFS

ONTAP supports parallel NFS (pNFS). The pNFS protocol offers performance improvements by giving clients direct access to the data of a set of files distributed across multiple nodes of a cluster. It helps clients locate the optimal path to a volume.

Use of hard mounts

When troubleshooting mounting problems, you need to be sure that you are using the correct mount type. NFS supports two mount types: soft mounts and hard mounts. You should use only hard mounts for reliability reasons.

You should not use soft mounts, especially when there is a possibility of frequent NFS timeouts. Race conditions can occur as a result of these timeouts, which can lead to data corruption.

NFS and SMB file and directory naming dependencies

Overview of NFS and SMB file and directory naming dependencies

File and directory naming conventions depend on both the network clients' operating systems and the file-sharing protocols, in addition to language settings on the ONTAP cluster and clients.

The operating system and the file-sharing protocols determine the following:

- Characters a file name can use
- Case-sensitivity of a file name

ONTAP supports multi-byte characters in file, directory, and qtree names, depending on the ONTAP release.

Characters a file or directory name can use

If you are accessing a file or directory from clients with different operating systems, you should use characters that are valid in both operating systems.

For example, if you use UNIX to create a file or directory, do not use a colon (:) in the name because the colon is not allowed in MS-DOS file or directory names. Because restrictions on valid characters vary from one operating system to another, see the documentation for your client operating system for more information about prohibited characters.

Case-sensitivity of file and directory names in a multiprotocol environment

File and directory names are case-sensitive for NFS clients and case-insensitive but case-preserving for SMB clients. You must understand what the implications are in a multiprotocol environment and the actions you might need to take when specifying the path while creating SMB shares and when accessing data within the shares.

If an SMB client creates a directory named `testdir`, both SMB and NFS clients display the file name as `testdir`. However, if an SMB user later tries to create a directory name `TESTDIR`, the name is not allowed

because, to the SMB client, that name currently exists. If an NFS user later creates a directory named `TESTDIR`, NFS and SMB clients display the directory name differently, as follows:

- On NFS clients, you see both directory names as they were created, for example `testdir` and `TESTDIR`, because directory names are case-sensitive.
- SMB clients use the 8.3 names to distinguish between the two directories. One directory has the base file name. Additional directories are assigned an 8.3 file name.
  - On SMB clients, you see `testdir` and `TESTDI~1`.
  - ONTAP creates the `TESTDI~1` directory name to differentiate the two directories.

    In this case, you must use the 8.3 name when specifying a share path while creating or modifying a share on a storage virtual machine (SVM).

Similarly for files, if an SMB client creates `test.txt`, both SMB and NFS clients display the file name as `text.txt`. However, if an SMB user later tries to create `Test.txt`, the name is not allowed because, to the SMB client, that name currently exists. If an NFS user later creates a file named `Test.txt`, NFS and SMB clients display the file name differently, as follows:

- On NFS clients, you see both file names as they were created, `test.txt` and `Test.txt`, because file names are case-sensitive.
- SMB clients use the 8.3 names to distinguish between the two files. One file has the base file name. Additional files are assigned an 8.3 file name.
  - On SMB clients, you see `test.txt` and `TEST~1.TXT`.
  - ONTAP creates the `TEST~1.TXT` file name to differentiate the two files.

If you have enabled or modified character mapping using the Vserver CIFS character-mapping commands, a normally case-insensitive Windows lookup becomes case-sensitive.

How ONTAP creates file and directory names

ONTAP creates and maintains two names for files or directories in any directory that has access from an SMB client: the original long name and a name in 8.3 format.

For file or directory names that exceed the eight character name or the three character extension limit (for files), ONTAP generates an 8.3-format name as follows:

- It truncates the original file or directory name to six characters, if the name exceeds six characters.
- It appends a tilde (~) and a number, one through five, to file or directory names that are no longer unique after being truncated.

  If it runs out of numbers because there are more than five similar names, it creates a unique name that bears no relation to the original name.

- In the case of files, it truncates the file name extension to three characters.

For example, if an NFS client creates a file named `specifications.html`, the 8.3 format file name created by ONTAP is `specif~1.htm`. If this name already exists, ONTAP uses a different number at the end of the file name. For example, if an NFS client then creates another file named `specifications_new.html`, the 8.3 format of `specifications_new.html` is `specif~2.htm`.

How ONTAP handles multi-byte file, directory, and qtree names

Beginning with ONTAP 9.5, support for 4-byte UTF-8 encoded names enables the creation and display of file, directory, and tree names that include Unicode supplementary characters outside the Basic Multilingual Plane (BMP). In earlier releases, these supplementary characters did not display correctly in multiprotocol environments.

To enable support for 4-byte UTF-8 encoded names, a new *utf8mb4* language code is available for the `vserver` and `volume` command families.

- You must create a new volume in one of the following ways:
- Setting the volume `-language` option explicitly:

  ```
  volume create -language utf8mb4 {…}
  ```

- Inheriting the volume `-language` option from an SVM that has been created with or modified for the option:

  ```
  vserver [create|modify] -language utf8mb4 {…}``volume create {…}
  ```

- You cannot modify existing volumes for utf8mb4 support; you must create a new utf8mb4-ready volume, and then migrate the data using client-based copy tools.

  You can update SVMs for utf8mb4 support, but existing volumes retain their original language codes.

  LUN names with 4-byte UTF-8 characters are not currently supported.

- Unicode character data is typically represented in Windows file systems applications using the 16-bit Unicode Transformation Format (UTF-16) and in NFS file systems using the 8-bit Unicode Transformation Format (UTF-8).

  In releases prior to ONTAP 9.5, names including UTF-16 supplementary characters that were created by Windows clients were correctly displayed to other Windows clients but were not translated correctly to UTF-8 for NFS clients. Similarly, names with UTF-8 supplementary characters by created NFS clients were not translated correctly to UTF-16 for Windows clients.

- When you create file names on systems running ONTAP 9.4 or earlier that contain valid or invalid supplementary characters, ONTAP rejects the file name and returns an invalid file name error.

  To avoid this issue, use only BMP characters in file names and avoid using supplementary characters, or upgrade to ONTAP 9.5 or later.

Unicode characters are allowed in qtree names.

- You can use either the `volume qtree` command family or System Manager to set or modify qtree names.
- qtree names can include multi-byte characters in Unicode format, such as Japanese and Chinese characters.
- In releases before ONTAP 9.5, only BMP characters (that is, those that could be represented in 3 bytes) were supported.

In releases before ONTAP 9.5, the junction-path of the qtree's parent volume can contain qtree and directory names with Unicode characters. The `volume show` command displays these names correctly when the parent volume has a UTF-8 language setting. However, if the parent volume language is not one of the UTF-8 language settings, some parts of the junction-path are displayed using a numeric NFS alternate name.

- In 9.5 and later releases, 4-byte characters are supported in qtree names, provided that the qtree is in a volume enabled for utf8mb4.

Configure character mapping for SMB file name translation on volumes

NFS clients can create file names that contain characters that are not valid for SMB clients and certain Windows applications. You can configure character mapping for file name translation on volumes to allow SMB clients to access files with NFS names that would otherwise not be valid.

About this task

When files created by NFS clients are accessed by SMB clients, ONTAP looks at the name of the file. If the name is not a valid SMB file name (for example, if it has an embedded colon ":" character), ONTAP returns the 8.3 file name that is maintained for each file. However, this causes problems for applications that encode important information into long file names.

Therefore, if you are sharing a file between clients on different operating systems, you should use characters in the file names that are valid in both operating systems.

However, if you have NFS clients that create file names containing characters that are not valid file names for SMB clients, you can define a map that converts the invalid NFS characters into Unicode characters that both SMB and certain Windows applications accept. For example, this functionality supports the CATIA MCAD and Mathematica applications as well as other applications that have this requirement.

You can configure character mapping on a volume-by-volume basis.

You must keep the following in mind when configuring character mapping on a volume:

- Character mapping is not applied across junction points.

  You must explicitly configure character mapping for each junction volume.

- You must make sure that the Unicode characters that are used to represent invalid or illegal characters are characters that do not normally appear in file names; otherwise, unwanted mappings occur.

  For example, if you try to map a colon (:) to a hyphen (-) but the hyphen (-) was used in the file name correctly, a Windows client trying to access a file named "a-b" would have its request mapped to the NFS name of "a:b" (not the desired outcome).

- After applying character mapping, if the mapping still contains an invalid Windows character, ONTAP falls back to Windows 8.3 file names.
- In FPolicy notifications, NAS audit logs, and security trace messages, the mapped file names are shown.
- When a SnapMirror relation of type DP is created, the source volume's character mapping is not replicated on the destination DP volume.
- Case sensitivity: Because the mapped Windows names turn into NFS names, the lookup of the names follows NFS semantics. That includes the fact that NFS lookups are case-sensitive. This means that the

applications accessing mapped shares must not rely on Windows case-insensitive behavior. However, the 8.3 name is available, and that is case-insensitive.

- Partial or invalid mappings: After mapping a name to return to clients doing directory enumeration ("dir"), the resulting Unicode name is checked for Windows validity. If that name still has invalid characters in it, or if it is otherwise invalid for Windows (e.g. it ends in "." or blank) the 8.3 name is returned instead of the invalid name.

Step

1. Configure character mapping:

   ```
   vserver cifs character-mapping create -vserver vserver_name -volume
   volume_name -mapping mapping_text, ...
   ```

   The mapping consists of a list of source-target character pairs separated by “:”. The characters are Unicode characters entered using hexadecimal digits. For example: 3C:E03C.

   The first value of each `mapping_text` pair that is separated by a colon is the hexadecimal value of the NFS character you want to translate, and the second value is the Unicode value that SMB uses. The mapping pairs must be unique (a one-to-one mapping should exist).

   - Source mapping

     The following table shows the permissible Unicode character set for source mapping:

Unicode character	Printed character	Description
0x01-0x19	Not applicable	Non-printing control characters
0x5C	\	Backslash
0x3A	:	Colon
0x2A	*	Asterisk
0x3F	?	Question mark
0x22	"	Quotation mark
0x3C	<	Less than
0x3E	>	Greater than
0x7C	|	Vertical line
0xB1	±	Plus-minus sign

   - Target mapping

     You can specify target characters in the “Private Use Area” of Unicode in the following range:

U+E0000…U+F8FF.

Example

The following command creates a character mapping for a volume named “data” on storage virtual machine (SVM) vs1:

```
cluster1::> vserver cifs character-mapping create -volume data -mapping
3c:e17c,3e:f17d,2a:f745
cluster1::> vserver cifs character-mapping show

Vserver         Volume Name  Character Mapping
--------------  -----------  ------------------------------------------
vs1             data         3c:e17c, 3e:f17d, 2a:f745
```

Commands for managing character mappings for SMB file name translation

You can manage character mapping by creating, modifying, displaying information about, or deleting file character mappings used for SMB file name translation on FlexVol volumes.

If you want to…	Use this command…
Create new file character mappings	`vserver cifs character-mapping create`
Display information about file character mappings	`vserver cifs character-mapping show`
Modify existing file character mappings	`vserver cifs character-mapping modify`
Delete file character mappings	`vserver cifs character-mapping delete`

For more information, see the man page for each command.

Copyright information

Trademark information