IBM System Storage N series

Clustered Data ONTAP 8.2 File Access Management Guide for NFS

SC27-6604-00

Contents

Preface

About this guide

This document applies to IBM N series systems running Data ONTAP, including systems with gateway functionality. If the terms *Cluster-Mode* or *clustered Data ONTAP* are used in this document, they refer to the Data ONTAP features and functionality designed for clusters, which are different from 7-Mode and prior Data ONTAP 7.1, 7.2, and 7.3 release families.

In this document, the term *gateway* describes IBM N series storage systems that have been ordered with gateway functionality. Gateways support various types of storage, and they are used with third-party disk storage systems—for example, disk storage systems from IBM, HP®, Hitachi Data Systems®, and EMC®. In this case, disk storage for customer data and the RAID controller functionality is provided by the back-end disk storage system. A gateway might also be used with disk storage expansion units specifically designed for the IBM N series models.

The term *filer* describes IBM N series storage systems that either contain internal disk storage or attach to disk storage expansion units specifically designed for the IBM N series storage systems. Filer storage systems do not support using third-party disk storage systems.

Supported features

IBM System Storage N series storage systems are driven by NetApp Data ONTAP software. Some features described in the product software documentation are neither offered nor supported by IBM. Please contact your local IBM representative or reseller for further details.

Information about supported features can also be found on the N series support website (accessed and navigated as described in *Websites* on page 10).

Websites

IBM maintains pages on the World Wide Web where you can get the latest technical information and download device drivers and updates. The following web pages provide N series information:

- A listing of currently available N series products and features can be found at the following web page:
  www.ibm.com/storage/nas/
- The IBM System Storage N series support website requires users to register in order to obtain access to N series support content on the web. To understand how the N series support web

content is organized and navigated, and to access the N series support website, refer to the following publicly accessible web page:

www.ibm.com/storage/support/nseries/

This web page also provides links to AutoSupport information as well as other important N series product resources.

- IBM System Storage N series products attach to a variety of servers and operating systems. To determine the latest supported attachments, go to the IBM N series interoperability matrix at the following web page:

  www.ibm.com/systems/storage/network/interophome.html
- For the latest N series hardware product documentation, including planning, installation and setup, and hardware monitoring, service and diagnostics, see the IBM N series Information Center at the following web page:

  publib.boulder.ibm.com/infocenter/nasinfo/nseries/index.jsp

Getting information, help, and service

If you need help, service, or technical assistance or just want more information about IBM products, you will find a wide variety of sources available from IBM to assist you. This section contains information about where to go for additional information about IBM and IBM products, what to do if you experience a problem with your IBM N series product, and whom to call for service, if it is necessary.

Before you call

Before you call, make sure you have taken these steps to try to solve the problem yourself:

- Check all cables to make sure they are connected.
- Check the power switches to make sure the system is turned on.
- Use the troubleshooting information in your system documentation and use the diagnostic tools that come with your system.
- Refer to the N series support website (accessed and navigated as described in *Websites* on page 10) for information on known problems and limitations.

Using the documentation

The latest versions of N series software documentation, including Data ONTAP and other software products, are available on the N series support website (accessed and navigated as described in *Websites* on page 10).

Current N series hardware product documentation is shipped with your hardware product in printed documents or as PDF files on a documentation CD. For the latest N series hardware product documentation PDFs, go to the N series support website.

Hardware documentation, including planning, installation and setup, and hardware monitoring, service, and diagnostics, is also provided in an IBM N series Information Center at the following web page:

publib.boulder.ibm.com/infocenter/nasinfo/nseries/index.jsp

Hardware service and support

You can receive hardware service through IBM Integrated Technology Services. Visit the following web page for support telephone numbers:

www.ibm.com/planetwide/

Firmware updates

IBM N series product firmware is embedded in Data ONTAP. As with all devices, ensure that you run the latest level of firmware. Any firmware updates are posted to the N series support website (accessed and navigated as described in *Websites* on page 10).

Note: If you do not see new firmware updates on the N series support website, you are running the latest level of firmware.

Verify that the latest level of firmware is installed on your machine before contacting IBM for technical support.

How to send your comments

Your feedback helps us to provide the most accurate and high-quality information. If you have comments or suggestions for improving this document, please send them by email to *starpubs@us.ibm.com*.

Be sure to include the following:

- Exact publication title
- Publication form number (for example, GC26-1234-02)
- Page, table, or illustration numbers
- A detailed description of any information that should be changed

Considerations before configuring file access

Data ONTAP allows you to manage access to files by clients using different protocols. There are certain concepts you should be familiar with before configuring file access.

File protocols that Data ONTAP supports

Data ONTAP supports file access using the NFS and CIFS protocols.

This means clients can access all files on Storage Virtual Machines (SVMs) regardless of what protocol they are connecting with or what type of authentication they require.

How Data ONTAP controls access to files

Data 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, Data ONTAP has to perform two tasks:

- Authentication
  Data 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
  Data 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, Data 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 Data 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).

Data ONTAP supports Kerberos authentication from both UNIX and Windows servers.

File-based restrictions

With file-based restrictions, you can specify which users can access which files.

When a user creates a file, Data ONTAP generates a list of access permissions for the file. Although the form of the permissions list varies with each protocol, it always includes common permissions, such as reading and writing permissions.

When a user tries to access a file, Data ONTAP uses the permissions list to determine whether to grant access. Data ONTAP grants or denies access according to the operation that the user is performing, such as reading or writing, and the following factors:

- User account
- User groups or netgroups
- Client protocol
- File type

As part of the verification process, Data ONTAP maps host names to IP addresses using the lookup service you specify—Lightweight Directory Access Protocol (LDAP), Network Information Service (NIS), Domain Name Service (DNS), or local storage system information.

LIF configuration requirements for file access management

To properly manage file access control, Data ONTAP must communicate with external services such as NIS, LDAP, and Active Directory servers. The Storage Virtual Machine (SVM) LIFs must be properly configured to allow these communications.

The communication with external services usually happens over the data LIF of the SVM. Therefore, you must ensure that the SVM has a data LIF properly configured to reach all required external services on each node.

In addition, in some situations, communication over the data LIF might fail or must be made on a node that does not host data LIFs for the SVM. In this case, the storage system attempts to use node-management and cluster-management LIFs instead. If your environment allows this, you should also ensure that the node-management and cluster-management LIFs in the cluster can reach these external services as well.

For more information about LIF configuration, see the *Clustered Data ONTAP Network Management Guide*.

How namespaces and volume junctions affect file access on SVMs with FlexVol volumes

You must understand what namespaces and volume junctions are and how they work to correctly configure file access on Storage Virtual Machines (SVMs) in your storage environment.

What namespaces in SVMs with FlexVol volumes are

A namespace is a logical grouping of volumes that are joined together at junction points to create a single, logical file system that derives from the Storage Virtual Machine (SVM) root volume. Each SVM has a namespace.

CIFS and NFS servers on a data SVM can store and access data across the namespace. Each client can access the entire namespace by mounting an export or accessing a single SMB share at the top of the namespace.

Alternatively, SVM administrators can create exports at each volume junction so that clients can create mount points at intermediate locations in the namespace, or they can create SMB shares that point to any directory path in the namespace.

Volumes can be added at any time by mounting them to any location in the namespace. Clients can immediately access the newly added volume, provided that the volume junction is under the point at which they are accessing the namespace and provided that they have sufficient permissions.

What volume junctions are

Volume junctions are a way to join individual volumes together into a single, logical namespace. Volume junctions are transparent to CIFS and NFS clients. When NAS clients access data by traversing a junction, the junction appears to be an ordinary directory.

A junction is formed when a volume is mounted to a mount point below the root and is used to create a file-system tree. The top of a file-system tree is always the root volume, which is represented by a slash (/). A junction points from a directory in one volume to the root directory of another volume.

A volume must be mounted at a junction point in the namespace to allow NAS client access to contained data:

- Although specifying a junction point is optional when a volume is created, data in the volume cannot be exported and a share cannot be created until the volume is mounted to a junction point in the namespace.
- A volume that was not mounted during volume creation can be mounted post-creation.
- New volumes can be added to the namespace at any time by mounting them to a junction point.
- Mounted volumes can be unmounted; however, unmounting a volume disrupts NAS client access to all data in the volume and to all volumes mounted at child junction points beneath the unmounted volume.

- Junction points can be created directly below a parent volume junction, or they can be created on a directory within a volume.

  For example, a path to a volume junction for a volume named "vol3" might be `/vol1/vol2/vol3`, or it might be `/vol1/dir2/vol3`, or even `/dir1/dir2/vol3`.

For more information, see the *Clustered Data ONTAP File Access Management Guide for CIFS* or the *Clustered Data ONTAP File Access Management Guide for NFS*.

How volume junctions are used in SMB and NFS namespaces

You can mount volumes at junction points anywhere within the namespace to create a single, logical namespace. If you specify a junction point when the volume is created, the volume is automatically mounted at the time the volume is created and is available for NAS access. You can create SMB shares and NFS exports on the mounted volume.

If you do not specify a junction point, the volume is online but is not mounted for NAS file access. You must mount a volume to a junction point before it can be used for NAS file access.

What the typical NAS namespace architectures are

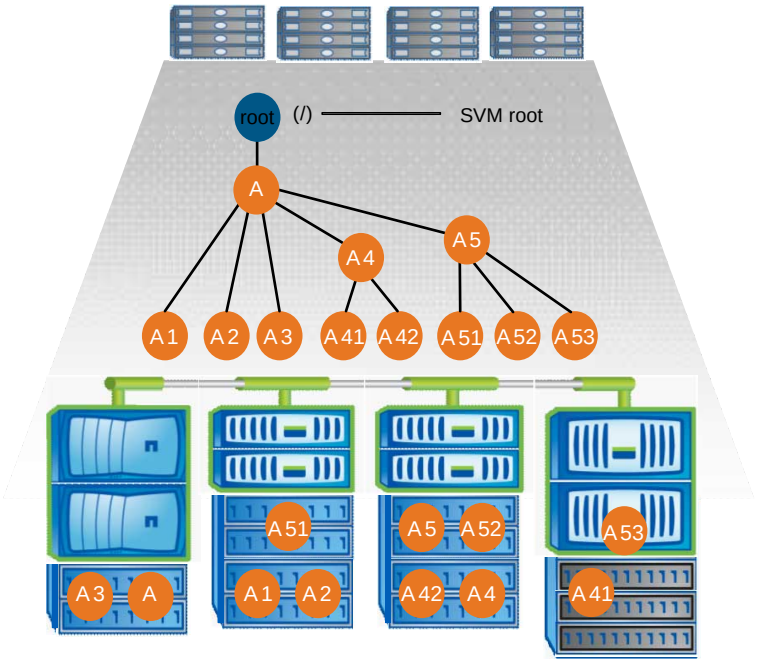
All Storage Virtual Machine (SVM) name spaces derive from the root volume; however, there are several typical NAS namespace architectures that you can use as you create your SVM name space. You can choose the namespace architecture that matches your business and workflow needs.

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).

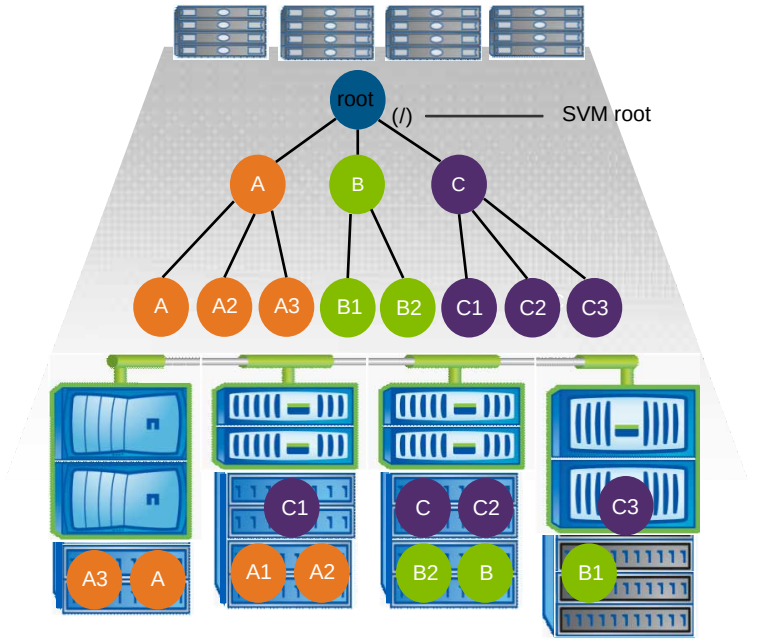


For example, a typical volume junction configuration with the above namespace architecture might look like the following configuration, where all volumes are junctioned below the single insertion point, which is a directory named “data”:

```
                 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).

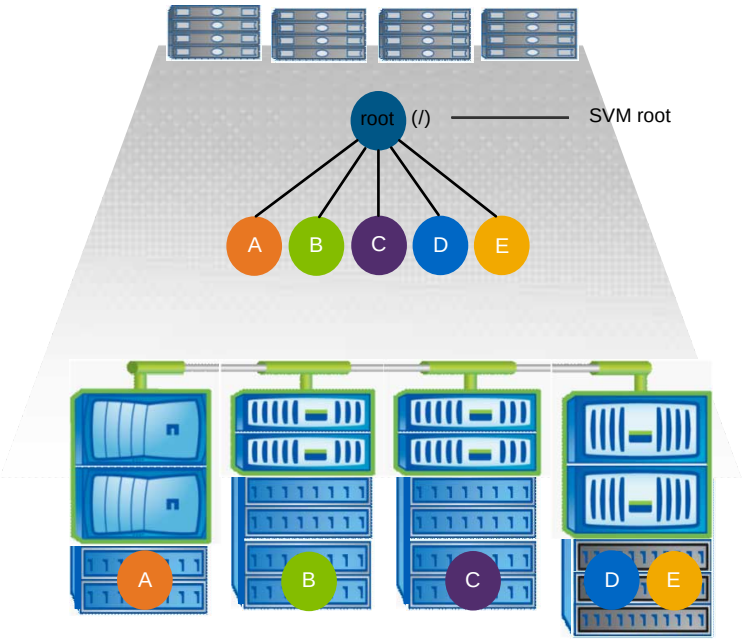


For example, a typical volume junction configuration with the above namespace architecture might look like the following configuration, where there are three insertion points to the root volume of the SVM. Two insertion points are directories named “data” and “projects”. One insertion point is a junctioned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

```
                                Junction
Vserver Volume                  Active   Junction Path          Junction
                                                                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 security styles affect data access

Each volume and qtree on the storage system has a security style. The security style determines what type of permissions are used for data on volumes when authorizing users. You must understand what the different security styles are, when and where they are set, how they impact permissions, how they differ between volume types, and more.

For more information about security styles, see the *Clustered Data ONTAP File Access Management Guide for CIFS* or *Clustered Data ONTAP File Access Management Guide for NFS*.

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 Data 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 Data ONTAP. However, Data 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	UNIX	
NTFS	SMB	NTFS ACLs	NTFS	
Mixed	NFS or SMB	NFSv3 mode bits	UNIX	
		NFSv4.x ACLs	UNIX	
		NTFS ACLs	NTFS	
Unified (only for Infinite Volumes)	NFS or SMB	NFSv3 mode bits	UNIX	
		NFSv4.1 ACLs	UNIX	
		NTFS ACLs	NTFS	

For more information about the unified security style, see the *Clustered Data ONTAP Infinite Volumes Management Guide*.

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.

Note: Data 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.

Note: Infinite Volumes always use the unified security style. You cannot configure or change the security style of an Infinite Volume.

How to decide on what security style to use on SVMs with FlexVol volumes

To help you decide what 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 qtree, it inherits its security style.

Security styles are inherited in the following manner:

- A FlexVol volume inherits the security style of the root volume of its containing Storage Virtual Machine (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.

Infinite Volumes cannot inherit security styles. All files and directories in Infinite Volumes always use the unified security style. The security style of an Infinite Volume and the files and directories it contains cannot be changed.

How Data ONTAP preserves UNIX permissions

When files in a FlexVol volume that currently have UNIX permissions are edited and saved by Windows applications, Data 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. Data ONTAP does not set any NTFS ACLs using the constructed ACL.

How to manage UNIX permissions using the Windows Security tab

If you want to manipulate UNIX permissions of files or folders in UNIX or mixed security-style qtrees or volumes on Storage Virtual Machines (SVMs) with FlexVol volumes, 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 UNIX security-style volume or qtree. This is also true for a mixed security-style volume or qtree where the files and folders have a UNIX effective security style.

  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 the listed entries and then 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 UNIX security objects and adding 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.

NFS and CIFS file naming dependencies

File naming conventions depend on both the network clients' operating systems and the file-sharing protocols.

The operating system and the file-sharing protocols determine the following:

- Characters a file name can use
- Case-sensitivity of a file name

Characters a file name can use

If you are sharing a file between clients on different operating systems, you should use characters that are valid in both operating systems.

For example, if you use UNIX to create a file, do not use a colon (:) in the file name because the colon is not allowed in MS-DOS file 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 a file name

File names are case-sensitive for NFS clients and case-insensitive but case-preserving for CIFS clients.

For example, if a CIFS client creates `Spec.txt`, both CIFS and NFS clients display the file name as `Spec.txt`. However, if a CIFS user later tries to create `spec.txt`, the name is not allowed because, to the CIFS client, that name currently exists. If an NFS user later creates a file named `spec.txt`, NFS and CIFS clients display the file name differently, as follows:

- On NFS clients, you see both file names as they were created, `Spec.txt` and `spec.txt`, because file names are case-sensitive.
- On CIFS clients, you see `Spec.txt` and `Spec~1.txt`.
  Data ONTAP creates the `Spec~1.txt` file name to differentiate the two files.

How Data ONTAP creates file names

Data ONTAP creates and maintains two file names for files in any directory that has access from a CIFS client: the original long name and a file name in 8.3 format.

For file names that exceed the eight character name or the three character extension limit, Data ONTAP generates an 8.3-format file name as follows:

- It truncates the original file name to six characters, if the file name exceeds six characters.
- It appends a tilde (~) and a number, one through five, to file 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 file name that bears no relation to the original file name.

- 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 Data ONTAP is `specif~1.htm`. If this name already exists, Data 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`.

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.

How Data ONTAP supports file access using NFS

You can export and unexport volumes or qtrees on your storage system, making them available or unavailable, respectively, for mounting by NFS clients.

How Data ONTAP handles NFS client authentication

NFS clients must be properly authenticated before they can access data on the Storage Virtual Machine (SVM). Data ONTAP authenticates the clients by checking their UNIX credentials against name services you configure.

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

How Data ONTAP grants CIFS file access from NFS clients

Data 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.

Data ONTAP does this by converting the user's UNIX User ID (UID) into a CIFS credential, and then using the CIFS credential to verify that the user has access rights to the file. A CIFS 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 Data ONTAP takes converting the UNIX UID into a CIFS credential can be from tens of milliseconds to hundreds of milliseconds because the process involves contacting a domain controller. Data ONTAP maps the UID to the CIFS credential and enters the mapping in a credential cache to reduce the verification time caused by the conversion.

Supported NFS versions and clients

Before you can use NFS in your network, you need to know which NFS versions and clients Data ONTAP supports.

For the latest information about which NFS versions and clients Data ONTAP supports, see the N series Interoperability Matrices website (accessed and navigated as described in *Websites* on page 10).

NFSv4.0 functionality supported by Data ONTAP

Data 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 Data ONTAP support for NFSv4

You should be aware of several limitations of Data ONTAP support for NFSv4.

- The SPKM3 and LIPKEY security mechanisms are not supported.
- The delegation feature is not supported by every client type.
- Names with non-ASCII characters on volumes other than UTF8 volumes are rejected by the storage system.
- 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.
  Data 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`

  Note: Although it does not support the `quota*` attributes, Data ONTAP does support user and group quotas through the RQUOTA side band protocol.

Data ONTAP support for NFSv4.1

Data ONTAP supports the NFSv4.1 protocol to allow access for NFSv4.1 clients.

By default NFSv4.1 is disabled. 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).

Data ONTAP does not support NFSv4.1 directory and file level delegations.

Data ONTAP support for parallel NFS

Data 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.

Where to find information about NFS support on Infinite Volumes

For information about the NFS versions and functionality that Infinite Volumes support, see the *Clustered Data ONTAP Infinite Volumes Management Guide*.

Process for NFS access to UNIX security style data on SVMs with FlexVol volumes

Understanding the process used for NFS access to UNIX security style data is helpful when designing a file access configuration that provides appropriate security settings.

When an NFS client connects to a storage system to access data with UNIX security style, Data ONTAP goes through the following steps:

1. Obtain the UNIX credentials for the user.
   Data ONTAP checks local UNIX accounts, NIS servers, and LDAP servers, depending on the Storage Virtual Machine (SVM) configuration.
2. Authorize the user.
   Data ONTAP checks the UNIX credentials for the user against the UNIX permissions of the data to determine what type of data access the user is allowed, if any.

In this scenario, name mapping is not performed because CIFS credentials are not required.

Process for NFS access to NTFS security style data on SVMs with FlexVol volumes

Understanding the process used for NFS access to NTFS security style data is helpful when designing a file access configuration that provides appropriate security settings.

When an NFS client connects to a storage system to access data with NTFS security style, Data ONTAP goes through the following steps:

1. Obtain the UNIX credentials for the user.
   Data ONTAP checks local UNIX accounts, NIS servers, and LDAP servers, depending on the Storage Virtual Machine (SVM) configuration.
2. Map the UNIX user to a CIFS name.
   Data ONTAP checks local name mapping rules, LDAP mapping rules, and the default CIFS user, depending on the SVM configuration.
3. Establish a connection to a Windows domain controller.
   Data ONTAP uses a cached connection, queries DNS servers, or uses a specified preferred domain controller.
4. Authenticate the user.
   Data ONTAP connects to the domain controller and performs pass-through authentication.
5. Authorize the user.

Data ONTAP checks the CIFS credentials for the user against the NTFS permissions of the data to determine what type of data access the user is allowed, if any.

Setting up file access using NFS

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 desired NFS protocol versions on the SVM.
   You can specify the versions of NFS that clients can use to access files on the SVM.
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.
3. Configure the NFS server with the appropriate security, name mapping, and other settings depending on the network and storage environment.

Note: The SVM must exist before you can set up file access using NFS. For more information about SVMs, see the *Clustered Data ONTAP System Administration Guide for SVM Administrators*

Creating and managing data volumes in NAS namespaces

To manage file access in a NAS environment, you must manage data volumes and junction points on your Storage Virtual Machine (SVM) with FlexVol volumes. This includes planning your namespace architecture, creating volumes with or without junction points, mounting or unmounting volumes, and displaying information about data volumes and NFS server or CIFS server namespaces.

Creating 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.

Before you begin

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

Steps

1. Create the volume with 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} -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, Data 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.

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
```

Creating 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.

Before you begin

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, Data 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     -        -               -
```

Mounting or unmounting 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 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. When you unmount 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 command...
Mount a volume	**volume mount -vserver *vserver_name* -volume *volume_name* -junction-path *junction_path***
Unmount a volume	**volume unmount -vserver *vserver_name* -volume *volume_name***

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

   volume show -vserver *vserver_name* -volume *volume_name* -junction

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 -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 unmounts a volume named “data” located on SVM vs1:

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

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

Displaying 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
```

Configuring security styles

You configure security styles on FlexVol volumes and qtrees to determine the type of permissions Data ONTAP uses to control access and what client type can modify these permissions.

For information about the security style of Infinite Volumes, see the *Clustered Data ONTAP Infinite Volumes Management Guide*.

Configuring 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. Perform one of the following actions:

Create the SVM with the...	Specify the security style by...
`vserver setup` command	Entering the desired root volume security style when prompted by the CLI wizard.
`vserver create` command	Including the `-rootvolume-security-style` parameter with the desired security style.

   The possible options for the root volume security style are `unix`, `ntfs`, or `mixed`. You cannot use `unified` security style because it only applies to Infinite Volumes.

   For more information about the `vserver setup` or `vserver create` commands, see the *Clustered Data ONTAP System Administration Guide for Cluster Administrators*.

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

   `vserver show -vserver *vserver_name*`

Configuring 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`. You cannot use `unified` security style because it only applies to Infinite Volumes.

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 the *Clustered Data ONTAP Logical Storage Management Guide*.

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`

Configuring 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`. You cannot use `unified` security style because it only applies to Infinite Volumes.

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 the *Clustered Data ONTAP Logical Storage Management Guide*.

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`

Modifying protocols for SVMs

Before you can configure and use NFS or SMB on Storage Virtual Machines (SVMs), you must enable the protocol. This is typically done during SVM setup, but if you did not enable the protocol during setup, you can enable it later by using the `vserver modify` command.

Steps

1. Check which protocols are currently enabled for the SVM by entering the following command:

   vserver show -vserver *vserver_name* -fields allowed-protocols

2. Modify the list of enabled protocols for the SVM by entering the following command:

   vserver modify vserver *vserver_name* -allowed-protocols *protocol_name*[,*protocol_name*,...]

   You must enter the complete list of protocols you want to be enabled on the SVM, including the protocols that are already enabled. Any protocol not specified with the command is automatically disabled and moved to the disallowed protocol list.

   You can also use the SVM setup wizard to modify protocols for the SVM by using the `vserver setup` command.

   See the man page for each command for more information.

3. Confirm that the allowed protocol list was updated correctly by entering the following command:

   vserver show -vserver *vserver_name* -fields allowed-protocols

Examples

The following command displays which protocols are currently enabled on the SVM named vs1.

```
vs1::> vserver show -vserver vs1 -fields allowed-protocols
vserver allowed-protocols
------- -----------------
vs1     nfs
```

The following command allows access over SMB by adding `cifs` to the list of enabled protocols on the SVM named vs1.

```
vs1::> vserver modify -vserver vs1 -allowed-protocols nfs,cifs
```

Enabling or disabling 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`**

Enabling or disabling 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. By default, NFSv4.0 is disabled.

About this task

NFSv4.0 is not supported for Storage Virtual Machines (SVMs) with Infinite Volume.

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`**

Enabling or disabling 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. By default, NFSv4.1 is disabled.

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**

Enabling or disabling parallel NFS

To enable or disable parallel NFS (pNFS), you can modify the `-v4.1-pnfs` option. By default pNFS is enabled.

Before you begin

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

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

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**

Creating an NFS server

The NFS server is necessary to provide NFS clients with access to the Storage Virtual Machine (SVM). You can use the `vserver nfs create` command to create an NFS server.

Before you begin

The cluster administrator might also want to configure an NTP server for the SVM. See the *Clustered Data ONTAP System Administration Guide for SVM Administrators* for more information.

Before creating an NFS server, you might want to configure an NIS domain for the SVM. If not, the NFS server uses local-users and local-groups.

Step

1. Use the `vserver nfs create` command to create an NFS server.

Example

The following command creates an NFS server on the SVM named vs1 with NFSv3 disabled, NFSv4.0 enabled, and NFSv4.0 ACLs enabled.

```
vs1::> vserver nfs create -vserver vs1 -v3 disabled -v4.0 enabled -
v4.0-acl enabled
```

Securing NFS access using export policies

You can use export policies to restrict NFS access to volumes or qtrees to clients that match specific parameters.

For information about how export policies affect Infinite Volumes, see the *Clustered Data ONTAP Infinite Volumes Management Guide*.

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 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 Data ONTAP.

Default export policy for SVMs with FlexVol volumes

Each Storage Virtual Machine (SVM) with FlexVol volumes has a default export policy that contains no rules. An export policy with rules must exist before clients can access data on the SVM, and each FlexVol volume contained in the SVM must be associated with an export policy.

When you create your SVM with FlexVol volumes, 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 with FlexVol volume, 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 or qtree 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 security type used by the client to authenticate, for example, Kerberos v5, NTLM, or AUTH_SYS.

If a rule specifies multiple criteria, and the client does not match one or more of them, the rule does not apply.

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.

How to handle 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`
- `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 one specified in the access parameter.	Gets access for that level with its own user ID.
Does not match 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.

If the client's security type...	Then the client...
Does not match 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 for 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.

How to handle 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, Data 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`.

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.

Creating an export policy

Before creating export rules, you must create an export policy to hold them. You can use the `vserver export-policy create` command to create an export policy.

Steps

1. To create an export policy, enter the following command:
   `vserver export-policy create -vserver` ***`vserver_name`*** **`-policyname`** ***`policy_name`***

   `-vserver` ***`vserver_name`*** specifies the Storage Virtual Machine (SVM) name.

   `-policyname` ***`policy_name`*** specifies the name of the new export policy. The policy name can be up to 256 characters long.
2. Assign the export policy to a volume or qtree by performing one of the following actions:

To assign the export policy to a...	Enter the command...
Volume	**volume modify -vserver *vserver_name* -volume *volume_name* -policy *export_policy_name***
Qtree	**volume qtree modify -vserver *vserver_name* -volume *volume_name* -qtree *qtree_name* -export-policy *export_policy_name***

Example

The following command creates an export policy named rs1 on the SVM named vs1:

```
vs1::> vserver export-policy create -vserver vs1 -policyname rs1
```

Adding a rule to an export policy

You can use the `vserver export-policy rule create` command to create an export rule for an export policy. This enables you to define client access to data.

Before you begin

The export policy you want to add the export rules to must already exist.

Step

1. To create an export rule, enter the following command:

   vserver export-policy rule create -vserver *vserver_name* -policyname *policy_name* -ruleindex *integer* -protocol {any|nfs3|nfs|cifs|nfs4|flexcache},... -clientmatch *text* -rorule {any|none|never|krb5|ntlm|sys},... -rwrule {any|none|never|krb5|ntlm|sys},... -anon *user_ID* -superuser {any|none|krb5|ntlm|sys},... -allow-suid {true|false} -allow-dev {true|false}

   `-vserver` *vserver_name* specifies the Storage Virtual Machine (SVM) name.

   `-policyname` *policy_name* specifies the name of the existing export policy to add the rule to.

   `-ruleindex` *integer* specifies the index number for the rule. Rules are evaluated according to their order in the list of index numbers; rules with lower index numbers are evaluated first. For example, the rule with index number 1 is evaluated before the rule with index number 2.

   `-protocol` {`any`|`nfs3`|`nfs`|`cifs`|`nfs4`|`flexcache`} specifies the access protocols that the export rule applies to. You can specify a comma-separated list of multiple access protocols for an export rule. If you specify the protocol as `any`, do not specify any other protocols in the list. If you do not specify an access protocol, the default value of `any` is used.

`-clientmatch` *`text`* specifies the client to which the rule applies. You can specify the match in any of the following formats:

Client match format	Example
Domain name preceded by the "." character	`.example.com`
Host name	`host1`
IPv4 address	`10.1.12.24`
IPv4 address with a subnet mask expressed as a number of bits	`10.1.12.10/4`
IPv4 address with a network mask	`10.1.16.0/255.255.255.0`
IPv6 address in dotted format	`::1.2.3.4`
IPv6 address with a subnet mask expressed as a number of bits	`ff::00/32`
A single netgroup with the netgroup name preceded by the @ character	`@netgroup1`

Note: Entering an IP address range, such as 10.1.12.10-10.1.12.70, is not allowed. Entries in this format are interpreted as a text string and treated as a host name. Entering an IPv6 address with a network mask, such as ff::12/ff::00, is not allowed.

When specifying individual IP addresses in export rules for granular management of client access, do not specify IP addresses that are dynamically (for example, DHCP) or temporarily (for example, IPv6) assigned. Otherwise, the client loses access when its IP address changes.

When specifying host or domain names, ensure that they can be properly resolved through DNS, both forward and reverse.

`-rorule` {`any`|`none`|`never`|`krb5`|`ntlm`|`sys`|} provides read-only access to clients that authenticate with the specified security types.

`-rwrule` {`any`|`none`|`never`|`krb5`|`ntlm`|`sys`|} provides read-write access to clients that authenticate with the specified security types.

Note: A client can only get read-write access for a specific security type if the export rule allows read-only access for that security type as well. If the read-only parameter is more restrictive for a security type than the read-write parameter, the client cannot get read-write access.

You can specify a comma-separated list of multiple security types for a rule. If you specify the security type as `any` or `never`, do not specify any other security types. Choose from the following valid security types:

- `any`

A matching client can access the data regardless of security type.

- `none`
  If listed alone, clients with any security type are granted access as anonymous. If listed with other security types, clients with a specified security type are granted access and clients with any other security type are granted access as anonymous.
- `never`
  A matching client cannot access the data regardless of security type.
- `krb5`
  A matching client can access the data if it is authenticated by Kerberos 5.
- `ntlm`
  A matching client can access the data if it is authenticated by CIFS NTLM.
- `sys`
  A matching client can access the data if it is authenticated by NFS AUTH_SYS.

`-anon` *`user_ID`* specifies a UNIX user ID or user name that is mapped to client requests that arrive with a user ID of 0 (zero), which is typically associated with the user name root. The default value is 65534, which is typically associated with the user name nobody.

`-superuser` {`any`|`none`|`krb5`|`ntlm`|`sys`|} provides superuser access to clients that authenticate with the specified security types.

Note: A client can only get superuser access for a specific security type if the export rule allows read-only access for that security type as well. If the read-only parameter is more restrictive for a security type than the superuser parameter, the client cannot get superuser access.

`-allow-suid` {`true`|`false`} specifies whether to allow access to set user ID (SUID) and set group ID (SGID). If this parameter is set to `true`, clients can modify the SUID or SGID of files, directories, and volumes. If this parameter is set to `false`, clients can modify the SUID or SGID of directories and volumes but not files. The default is `true`.

`-allow-dev` {`true`|`false`} specifies whether to allow creation of devices. The default is `true`.

Examples

The following command creates an export rule on the SVM named vs1 in an export policy named rs1. The rule has the index number 1. The rule matches all clients. The rule enables all NFS access. It enables read-only access by all clients and requires Kerberos authentication for read-write access. Clients with the UNIX user ID 0 (zero) are mapped to user ID 65534 (which typically maps to the user name nobody). The rule enables SUID and SGID modification but does not enable the creation of devices.

```
vs1::> vserver export-policy rule create -vserver vs1
-policyname rs1 -ruleindex 1 -protocol nfs -clientmatch 0.0.0.0/0
-rorule any -rwrule krb5 -anon 65534 -allow-suid true -allow-dev
false
```

The following command creates an export rule on the SVM named vs2 in an export policy named expol2. The rule has the index number 21. The rule matches clients to members of the netgroup dev_netgroup_main. The rule enables all NFS access. It enables read-only access for clients that authenticated with AUTH_SYS and requires Kerberos authentication for read-write access. Clients with the UNIX user ID 0 (zero) are mapped to user ID 65534 (which typically maps to the user name nobody). The rule enables SUID and SGID modification but does not enable the creation of devices.

```
vs1::> vserver export-policy rule create -vserver vs2
-policyname expol2 -ruleindex 21 -protocol nfs -clientmatch
@dev_netgroup_main -rorule sys -rwrule krb5 -anon 65534
-allow-suid true -allow-dev false
```

Setting an export rule's index number

You can use the `vserver export-policy rule setindex` command to manually set an existing export rule's index number. This enables you to rearrange the order in which Data ONTAP processes export rules.

About this task

If the new index number is already in use, the command inserts the rule at the specified spot and reorders the list accordingly.

Step

1. To modify the index number of a specified export rule, enter the following command:

   vserver export-policy rule setindex -vserver *virtual_server_name* -policyname *policy_name* -ruleindex *integer* -newruleindex *integer*

   `-vserver` *`virtual_server_name`* specifies the Storage Virtual Machine (SVM) name.

   `-policyname` *`policy_name`* specifies the policy name.

   `-ruleindex` *`integer`* specifies the current index number of the export rule. If the specified index number is already in use by an existing export policy rule, the index numbers of that and all successive export policy rules are incremented by one.

   `-newruleindex` *`integer`* specifies the new index number of the export rule.

Example

The following command changes the index number of an export rule at index number 3 to index number 2 in an export policy named rs1 on the SVM named vs1.

```
vs1::> vserver export-policy rule setindex -vserver vs1
-policyname rs1 -ruleindex 3 -newruleindex 2
```

Associating an export policy to a FlexVol volume

Each FlexVol volume contained in the Storage Virtual Machine (SVM) must be associated with an export policy that contains export rules for clients to access data in the volume.

About this task

When you create the SVM, Data ONTAP creates a default export policy called *default* for the SVM. Data ONTAP assigns the *default* export policy to the SVM volumes. You can associate another custom export policy that you create instead of the default policy to a volume. Before you associate a custom export policy to a volume, you must create one or more export rules that allow the desired access to data in the volume and assign those export rules to the export policy that you want to associate with the volume.

You can associate an export policy to a volume when you create the volume or at any time after you create the volume. You can associate one export policy to the volume.

Steps

1. Use either the `volume create` or `volume modify` command with the `-policy` option to associate an export policy to the volume.

   Example

   ```
   cluster::> volume create -vserver vs1 -volume vol1
   -aggregate aggr2 -state online -policy cifs_policy -security-style
   ntfs
   -junction-path /dept/marketing -size 250g -space-guarantee volume
   ```

   For more information about the `volume create` and `volume modify` commands, see the man pages.

2. Verify that the export policy associated with the volume has the desired access configuration using the `vserver export-policy rule show` command with the `-policyname` option.

   Example

   ```
   cluster::> vserver export-policy rule show -policyname cifs_policy
   Virtual      Policy             Rule    Access   Client       RO
   Server       Name               Index   Protocol Match        Rule
   ------------ ------------------ ------  -------- ------------ ------
   vs1          cifs_policy        1       cifs      0.0.0.0/0    any
   ```

For more information about the `vserver export-policy rule show` command, see the man pages.

The command displays a summary for that export policy, including a list rules applied to that policy. Data ONTAP assigns each rule a rule index number. After you know the rule index number, you can use it to display detailed information about the specified export rule.

3. Verify that the rules applied to the export policy are configured correctly by using the `vserver export-policy rule show` command and specify the `-policyname`, `-vserver`, and `-ruleindex` options.

Example

```
cluster::> vserver export-policy rule show -policyname cifs_policy -
vserver vs1 -ruleindex 1
                             Virtual Server: vs1
                                Policy Name: cifs_policy
                                 Rule Index: 1
                            Access Protocol: cifs
Client Match Hostname, IP Address, Netgroup, or Domain: 0.0.0.0/0
                             RO Access Rule: any
                             RW Access Rule: any
User ID To Which Anonymous Users Are Mapped: 0
                  Superuser Security Flavors: any
                Honor SetUID Bits In SETATTR: true
                   Allow Creation of Devices: true
```

Assigning an export policy to a qtree

Instead of exporting an entire volume, you can also export a specific qtree on a volume to make it directly accessible to clients. You can export a qtree by assigning an export policy to it. You can assign the export policy either when you create a new qtree or by modifying an existing qtree.

Before you begin

The export policy must exist.

About this task

By default, qtrees use the parent export policy of the containing volume if not otherwise specified at the time of creation.

Steps

1. Perform one of the following actions:

To assign an export policy to...	Enter the command...
A new qtree	**volume qtree create -vserver *vserver_name* -qtree-path /vol/*volume_name*/*qtree_name* -export-policy *export_policy_name***
An existing qtree	**volume qtree modify -vserver *vserver_name* -qtree-path /vol/*volume_name*/*qtree_name* -export-policy *export_policy_name***

2. Verify that the export policy was assigned correctly by entering the following command:

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

Examples

The following command creates a new qtree named dev1 on volume vol_eng on the SVM named vs1 and assigns it the export policy dev1_export.

```
vs1::> volume qtree create -vserver vs1 -qtree-path /vol/vol_eng/
dev1 -export-policy dev1_export
```

The following command modifies an existing qtree named dev2 on volume vol_eng on the SVM named vs1 and assigns it the export policy dev2_export.

```
vs1::> volume qtree modify -vserver vs1 -qtree-path /vol/vol_eng/
dev2 -export-policy dev2_export
```

Removing 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 ("").

About this task

You must perform this task to remove all export policies assigned to qtrees before downgrading to a Data ONTAP release earlier than 8.2.1 that does not support qtree exports.

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 by entering the following command:
   `volume qtree show -qtree qtree_name -fields export-policy`

Validating qtree IDs for qtree file operations

Data 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

You can use Kerberos to provide strong authentication between Storage Virtual Machines (SVMs) and NFS clients to ensure secure NFS communication.

Beginning with version 3, NFS supports generic security services for RPC (RPCSEC_GSS), which enables the use of Kerberos 5. Kerberos provides strong secure authentication for client/server

applications. Authentication provides verification of a user's or process's identity to a server. In the Data ONTAP environment, Kerberos provides authentication between SVMs and NFS clients.

Group ID limitation for NFS RPCSEC_GSS

Although Data ONTAP supports generic security services for RPC (RPCSEC_GSS) for NFSv3 and later, you should be aware of a limitation related to the number of group IDs for UNIX users.

Data ONTAP supports up to 32 group IDs when handling NFS user credentials using RPCSEC_GSS authentication. If a user has more than 32 group IDs in their credentials, the remaining group IDs are truncated.

To avoid authentication issues due to this limitation, ensure that users do not belong to more than 32 groups.

Requirements for configuring Kerberos with NFS

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

Note: 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.

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.
- DES encryption

  The domain controller must have DES encryption enabled if you are running Windows Active Directory.
- 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 Storage Virtual Machine (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. For more information, see the N series Interoperability Matrices website (accessed and navigated as described in *Websites* on page 10).
- Kerberos

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

  - Encryption for TGS communication is enabled.

    DES for Windows Active Directory based Kerberos or MIT Kerberos, or DES3 for MIT Kerberos only.
  - The most secure encryption type for TGT communication is enabled.
  - The Kerberos realm and domain are configured correctly.
  - GSS is enabled.
- 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 DES-CBC-MD5 (for Windows Active Directory based Kerberos or MIT Kerberos) or DES3-CBC-SHA1 (for MIT Kerberos only). For Windows, you must allow weak cryptography to use DES.
- 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

- IPv4

  You must have IPv4 network connectivity configured. Kerberos is not supported over IPv6.
- NFS license

  The storage system must have a valid NFS license installed. For more information about managing feature licenses, see the *Clustered Data ONTAP System Administration Guide for Cluster Administrators*.
- 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. For more

information about managing feature licenses, see the *Clustered Data ONTAP System Administration Guide for Cluster Administrators*.

- SVM
  You must have at least one SVM configured on the storage system. For more information about configuring SVMs, see the *Clustered Data ONTAP Software Setup Guide*.
- DNS on the SVM
  You must have configured DNS on each SVM. For more information about configuring DNS on SVMs, see the *Clustered Data ONTAP Software Setup Guide*.
- NFS server
  You must have configured NFS on the SVM.
- CIFS server
  If you are running a multiprotocol environment, you must have configured CIFS on the SVM. The CIFS 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. For more information about configuring volumes, see the *Clustered Data ONTAP Logical Storage Management Guide*.
- Root volume
  The root volume of the SVM must have the following configuration:

Set the...	**To...**
Security style	UNIX
UID	root or ID 0
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 Data 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 server 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 Data 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 that is bound to the data LIF.

- 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 for the root volume to `anon=0`, `-rorule`, `-rwrule`, `-superuser`, and `-krb`.
- Kerberos-UNIX name mapping
  The SVM must have the Kerberos principal nfs/*`fqdn`*@*`REALM`* mapped to the UNIX user root.

Specifying 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, Data 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`***

Creating a Kerberos realm configuration

If you want to use Kerberos authentication for client access, you must first configure the Storage Virtual Machine (SVM) to use an existing Kerberos realm. You can use the `vserver services kerberos-realm create` command to configure the SVM to use a Kerberos realm.

Before you begin

The cluster administrator should have configured NTP on the storage system to avoid authentication issues. Time differences between a client and server (clock skew) are a common cause of authentication failures.

Step

1. Use the `vserver services kerberos-realm create` command to configure a Kerberos realm.

Examples

The following command creates a Kerberos realm configuration that uses a Microsoft Active Directory server as the KDC server. The Kerberos configuration is named AUTH. The Kerberos realm is AUTH.EXAMPLE.COM. The Active Directory server is named ad-1 and its IP address is 10.10.8.14. The permitted clock skew is 300 seconds (the default). The IP address of the KDC server is 10.10.8.14, and its port number is 88 (the default). The encryption type is Data Encryption Standard (DES). The comment is "Microsoft Kerberos config".

```
vs1::> vserver services kerberos-realm create -configname AUTH
-realm AUTH.EXAMPLE.COM -adserver-name ad-1 -adserver-ip 10.10.8.14
-clock-skew 300 -kdc-ip 10.10.8.14 -kdc-port 88 -kdc-vendor
Microsoft -comment "Microsoft Kerberos config"
```

The following command creates a Kerberos realm configuration that uses an MIT KDC. The Kerberos configuration is named SECURE. The Kerberos realm is SECURITY.EXAMPLE.COM. The permitted clock skew is 300 seconds and the Active Directory server and IP address are SUSAN and 10.10.3.1, respectively. The IP address of the KDC server is 10.10.9.1, and its port number is 88. The KDC vendor is Other to indicate a UNIX vendor. The IP address of the administrative server is 10.10.9.1, and its port number is 749 (the default). The IP address of the password server is 10.10.9.1, and its port number is 464 (the default). The encryption type is DES. The comment is "UNIX Kerberos config".

```
vs1::> vserver services kerberos-realm create -configname SECURE
-realm SECURITY.EXAMPLE.COM. -clock-skew 300 -adserver-name SUSAN -
adserver-ip 10.10.3.1
-kdc-ip 10.10.9.1 -kdc-port 88 -kdc-vendor Other -adminserver-ip
10.10.9.1 -adminserver-port 749
```

```
-passwordserver-ip 10.10.9.1 -passwordserver-port 464 -comment
"UNIX Kerberos config"
```

Creating an NFS Kerberos configuration

You can use the `vserver nfs kerberos-config modify` command to enable Kerberos and create a Kerberos configuration for Storage Virtual Machines (SVMs). This enables the SVM to use Kerberos security services for NFS.

About this task

For best performance, the SVM should have at least two data LIFs. One for use by the SPN for UNIX and CIFS-related Kerberos traffic, and another one for non-Kerberos traffic.

If you are using Microsoft Active Directory Kerberos, the first 15 characters of any service principal names (SPNs) used in the domain must be unique. Microsoft Active Directory (AD) has a limitation for SPNs of 15 characters maximum and does not allow duplicate SPNs.

Step

1. Enter the following command:

   vserver nfs kerberos-config modify -vserver *vserver_name* -lif *logical_interface* -kerberos {enabled|disabled} -spn *service_principal_name*

   If you need to create the SPN in a different OU of the Kerberos realm, you can specify the optional `-ou` parameter.

   See the man page for the command for more information.

Example

The following example creates an NFS Kerberos configuration for the SVM named vs1 on the logical interface ves03-d1 with the SPN nfs/ves03-d1.lab.example.com@TEST.LAB.EXAMPLE.COM in the OU lab2ou.

```
vs1::> vserver nfs kerberos-config modify -lif ves03-d1 -vserver
vs2 -kerberos enabled -spn nfs/ves03-
d1.lab.example.com@TEST.LAB.EXAMPLE.COM -ou "ou=lab2ou"
```

Configuring SVMs to use LDAP

You can configure Storage Virtual Machines (SVMs) to use LDAP servers in your environment for obtaining network information. You modify the `-ns-switch` parameter to configure the network

information sources and the `-nm-switch` parameter to configure the name mapping sources for SVMs.

Step

1. Perform one of the following actions:

If you want to configure SVMs to use LDAP as a...	Enter the command...
Network information source	**`vserver modify -vserver vserver_name -ns-switch file,ldap`**
Network information source and for name mapping.	**`vserver modify -vserver vserver_name -ns-switch file,ldap -nm-switch file,ldap`**

`-ns-switch` specifies the sources to search for network information and in what order.

`-nm-switch` specifies the sources to search for name mapping information and in what order. In a UNIX-only environment, this switch is not necessary. Name mapping is only required in a mixed environment using both UNIX and Windows.

See the man page for the command for more information.

Using LDAP over SSL/TLS to secure communication

You can use LDAP over SSL/TLS to secure communication between the Storage Virtual Machine (SVM) LDAP client and the LDAP server. This allows LDAP to encrypt all traffic to and from the LDAP server.

LDAP over SSL/TLS concepts

You must understand certain terms and concepts about how Data ONTAP uses SSL/TLS to secure LDAP communication. Data ONTAP can use LDAP over SSL/TLS 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 Data ONTAP uses LDAP over SSL to secure LDAP communication.

LDAP (Lightweight Directory Access Protocol) A set of protocols for accessing and managing information directories. LDAP is used as 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 secure protocol developed for sending information securely over the Internet. SSL is used to provide either server or mutual (server and client) authentication. SSL provides encryption only. If a method to ensure data integrity is needed, it must be provided by the application using SSL.
TLS	(Transport Layer Security) An IETF standards track protocol that is based on the earlier SSL specifications. It is the successor to SSL.
LDAP over SSL/TLS	(Also known as *LDAPS*) A protocol that uses SSL or TLS to secure communication between LDAP clients and LDAP servers. The terms *SSL* and *TLS* are often used interchangeably unless referring to a specific version of the protocol.
Start TLS	(Also known as *start_tls*, *STARTTLS*, and *StartTLS*) A mechanism to provide secure communication by using the TLS/SSL protocols.

How Data ONTAP uses LDAP over SSL/TLS

By default, LDAP communications between client and server applications are not encrypted. This means that it is possible to use a network monitoring device or software and view the communications between LDAP client and server computers. This is especially problematic when an LDAP simple bind is used because the credentials (user name and password) used to bind the LDAP client to the LDAP server are passed over the network unencrypted.

The SSL and TLS protocols run above TCP/IP and below higher-level protocols, such as LDAP. They use TCP/IP on behalf of the higher-level protocols, and in the process, permit an SSL-enabled server to authenticate itself to an SSL-enabled client and permit both machines to establish an encrypted connection. These capabilities address fundamental security concerns about communication over the Internet and other TCP/IP networks. Data ONTAP uses the START TLS method to set up the secured connection.

Data ONTAP supports SSL server authentication, which enables the Storage Virtual Machine (SVM) LDAP client to confirm the LDAP server's identity during the bind operation. SSL/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.

This version of Data ONTAP supports the following:

- LDAP over SSL/TLS for SMB-related traffic between the Active Directory-integrated LDAP servers and the SVM
- LDAP over SSL/TLS for LDAP traffic for name mapping
  Either Active Directory-integrated LDAP servers or UNIX-based LDAP servers can be used to store information for LDAP name mapping.
- 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.

Data ONTAP does not support signing (integrity protection) and sealing (encryption) of the data.

The default is not to enable LDAP over SSL/TLS.

Installing the self-signed root CA certificate on the SVM

Before you can use secure LDAP authentication when binding to LDAP servers, you must install the self-signed root CA certificate on the Storage Virtual Machine (SVM).

Steps

1. Install the self-signed root CA certificate:
   a) Enter the following command to begin the certificate installation:

      `security certificate install -vserver vserver_name -type server-ca`

      The console output displays the following message:
      `Please enter Certificate: Press <Enter> when done`
   b) Open the certificate `.pem` file with a text editor, copy the certificate, including the lines beginning with `-----BEGIN CERTIFICATE-----` and ending with `-----END CERTIFICATE-----`, and paste the certificate on the console.
   c) Verify that the certificate is displayed after the console prompt.
   d) To complete the installation, press **Enter**.
2. Verify that the certificate is installed:

   `security certificate show -vserver vserver_name`

Creating a new LDAP client schema

Data ONTAP provides three LDAP schemas: one for Active Directory Services for UNIX compatibility, one for Active Directory Identity Management for UNIX compatibility, and one for RFC-2307 LDAP compatibility. If the LDAP schema in your environment differs from these, you must create a new LDAP client schema for Data ONTAP.

About this task

The default LDAP schemas provided by Data ONTAP cannot be modified. To create a new schema, you create a copy and then modify the copy accordingly.

Steps

1. Display the existing LDAP client schema templates to identify the one you want to copy:

   `vserver services ldap client schema show`
2. Set the privilege level to advanced:

   `set -privilege advanced`
3. Make a copy of an existing LDAP client schema:

```
vserver services ldap client schema copy -vserver vserver_name -schema
existing_schema_name -new-schema-name new_schema_name
```

4. Use the `vserver services ldap client schema modify` command to modify the new schema and customize it for your environment.

   See the man page for the command for more information.

5. Return to the admin privilege level:

   ```
   set -privilege admin
   ```

Creating an LDAP client configuration

You can use the `vserver services ldap client create` command to create an LDAP client configuration. You must set up an LDAP client first to be able to use LDAP services.

Steps

1. To create an LDAP client configuration, enter the following command:

   ```
   vserver services ldap client create -vserver vserver_name -client-config
   client_config_name {-servers LDAP_server_list | -ad-domain ad_domain -
   preferred-ad-servers preferred_ad_server_list -bind-as-cifs-server
   {true|false}} -schema schema -port port -query-timeout integer -min-
   bind-level {anonymous|simple|sasl} -bind-dn LDAP_DN -bind-password
   password -base-dn LDAP_DN -base-scope {base|onelevel|subtree}
   ```

   `-vserver` *vserver_name* specifies the name of the SVM for which you want to create an LDAP client configuration.

   `-client-config` *client_config_name* specifies the name of the new LDAP client configuration.

   `-servers` *LDAP_server_list* specifies one or more LDAP servers by IP address in a comma-delimited list.

   `-ad-domain` *ad_domain* specifies the AD domain.

   `-preferred-ad-servers` *preferred_ad_server_list* specifies one or more preferred Active Directory servers by IP address in a comma-delimited list.

   `-bind-as-cifs-server` {`true`|`false`} specifies whether to bind using the CIFS credentials of the SVM. The default is `false`. If you want to set this parameter to `true`, you must first install a CIFS license and create a CIFS server.

   `-schema` *schema* specifies the schema template to use. You can either use one of the provided default schemas or create your own schema by copying a default schema (they are read-only) and modifying the copy.

   `-port` *port* specifies the LDAP server port. The default is `389`.

   `-query-timeout` *integer* specifies the query timeout in seconds. The allowed range is 0-10 seconds. The default is 3 seconds.

`-min-bind-level` {`anonymous`|`simple`|`sasl`} specifies the minimum bind authentication level. The default is `anonymous`.

`-bind-dn` *`LDAP_DN`* specifies the Bind user. For Active Directory servers, specify the user in the account (DOMAIN\user) or principal (user@domain.com) form. Otherwise, specify the user in distinguished name (CN=user,DC=domain,DC=com) form.

`-bind-password` *`password`* specifies the Bind password.

`-base-dn` *`LDAP_DN`* specifies the base DN. The default is `""` (root).

`-base-scope` {`base`|`onelevel`|`subtree`} specifies the base search scope. The default is `subtree`.

`-use-start-tls` {`true`|`false`} specifies whether to use Start TLS for the LDAP connection. If set to `true` and the LDAP server supports it, the LDAP client uses an encrypted TLS/SSL connection to the server. The default is `false`. You must install a self-signed root CA certificate of the LDAP server to use this option.

2. Verify that the LDAP client configuration was created successfully:

   `vserver services ldap client show -client-config client_config_name`

Examples

The following command creates a new LDAP client configuration named “ldap1” to work with an Active Directory server for LDAP:

```
cluster1::> vserver services ldap client create -vserver vs1 -
client-config ldapclient1 –ad-domain addomain.example.com -bind-as-
cifs-server true -schema AD-SFU -port 389 -query-timeout 3 -min-
bind-level simple -base-dn DC=addomain,DC=example,DC=com -base-
scope subtree -preferred-ad-servers 172.17.32.100
```

The following command modifies the LDAP client configuration named “ldap1” to work with an Active Directory server for LDAP by specifying an Active Directory domain and to bind to the server using the CIFS credentials of the SVM:

```
cluster1::> vserver services ldap client modify -vserver vs1 -
client-config ldap1 -bind-as-cifs-server true -ad-domain
addomain.example.com
```

The following command modifies the LDAP client configuration named “ldap1” by specifying the base DN:

```
cluster1::> vserver services ldap client modify -vserver vs1 -
client-config ldap1 -base-dn CN=Users,DC=addomain,DC=example,DC=com
```

Enabling LDAP on SVMs

You can use the `vserver services ldap create` command to enable LDAP on Storage Virtual Machines (SVMs) and configure it to use an LDAP client you created. This enables SVMs to use LDAP for name mapping.

Before you begin

An LDAP client configuration must exist on the SVM.

Step

1. To enable LDAP on SVMs, enter the following command:
   `vserver services ldap create -vserver *vserver_name* -client-config *client_config_name* -client-enabled true`

   `-vserver` *`vserver_name`* specifies the SVM name.

   `-client-config` *`client_config_name`* specifies the client configuration name.

   `-client-enabled` specifies whether the LDAP client is enabled. The default is `true`.

Example

The following command enables LDAP on the SVM named “vs1” and configures it to use the LDAP client configuration named “ldap1”.

```
cluster1::> vserver services ldap create -vserver vs1 -client-
config ldap1 -client-enabled true
```

How name mappings are used

Data ONTAP uses name mapping to map CIFS identities to UNIX identities, Kerberos identities to UNIX identities, and UNIX identities to CIFS 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 CIFS client.

Name mapping is usually required due to the multiprotocol nature of Data ONTAP, which supports CIFS and NFS client access to the same data. Data stored on Storage Virtual Machines (SVMs) with FlexVol volumes uses either UNIX- or NTFS-style permissions. To authorize a client, the credentials must match the security style. Consider the following scenarios:

If a CIFS client wants to access data with UNIX-style permissions, Data ONTAP cannot directly authorize the client because it cannot use CIFS credentials with UNIX-style permissions. To properly authorize the client request, Data ONTAP must first map the CIFS credentials to the appropriate UNIX credentials so that it can then use the UNIX credentials to compare them to the UNIX-style permissions.

If an NFS client wants to access data with NTFS-style permissions, Data ONTAP cannot directly authorize the client because it cannot use UNIX credentials with NTFS-style permissions. To properly authorize the client request, Data ONTAP must first map the UNIX credentials to the appropriate NTFS credentials so that it can then use the NTFS credentials to compare them to the NTFS-style permissions.

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 CIFS access or NTFS security style on volumes.

  In this scenario, name mapping is not required because Data ONTAP can use the UNIX credentials of the NFS clients to directly compare them to the UNIX-style permissions.
- 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. It is not possible to map CIFS users to a group ID (GID), or UNIX users to a group in the Active Directory (AD). Similarly, it is not possible to map a GID to a group or a user in AD, or an AD group to a UNIX UID or GID.

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. As a result, you can rename certain users in AD and use regular expressions to effectively emulate group actions. This type of mapping also works in reverse.

How name mapping works

Data ONTAP goes through a number of steps when attempting to map user names. They include checking the local name mapping database and LDAP, trying the user name, and using the default user if configured.

When Data 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 `-nm-switch` parameter of the Storage Virtual Machine (SVM) configuration.

- For Windows to UNIX mapping

  If no mapping is found, Data 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 it is configured. If the default UNIX user is not configured and it cannot obtain a mapping this way either, mapping fails and an error is returned.
- For UNIX to Windows mapping

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

Multidomain searches for UNIX user to Windows user name mappings

Data 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 Data ONTAP. Active Directory trust relationships with the CIFS 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 CIFS server on the Storage Virtual Machine (SVM) belongs.

- *Bidirectional trust*

  With bidirectional trusts, both domains trust each other. If the CIFS 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 CIFS 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.

Pattern	Replacement	Result
*	**	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.

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 Data ONTAP
  - The advantage to this method is that there is no management overhead and that the list is made of trusted domains that Data ONTAP has determined are valid.
  - The disadvantage is that you cannot choose the order that the trusted domains are searched.
- Use the preferred trusted domain list that you compile
  - The advantage to this method is that you can configure the list of trusted domains in the order that you want them searched.
  - The disadvantage is that there is more management overhead and that the list might become outdated, with some listed domains not being valid, bidirectionally trusted domains.

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.

  The search ends as soon as the Windows user is found. If the same Windows user name exists in two different trusted domains, then the user belonging to the domain listed first in the preferred trusted-domain list is returned. If the Windows user is not found in any domains in the preferred list, an error is returned.

  If you want the home domain to be included in the search, it must be included in the preferred trusted domain list.
- 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.

  The search ends as soon as the Windows user is found. If the same Windows user name exists in two different trusted domains, the user belonging to the domain listed first in the automatically discovered trusted-domain list is returned. You cannot control the order of the trusted domains in the automatically discovered list. If the Windows user is not found in any of the discovered trusted domains, the user is then looked up in the home domain.
- If there are no bidirectionally trusted domains for the home domain, the user is looked up in the home domain.

For more information about managing the list of bidirectional trusted domains, see the *Clustered Data ONTAP File Access Management Guide for CIFS*.

Name mapping conversion rules

A Data ONTAP system keeps a set of conversion rules for each Storage Virtual Machine (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.

It is possible to allow NFS access to volumes with NTFS security style for users in a different domain from the one that the storage system belongs to, provided that the proper name mapping rule exists.

If a user matches a rule to map to a user in a different domain, the domain must be trusted. To ensure successful mapping to users in other domains for both SMB and NFS access, there must be a bidirectional trust relationship between the domains.

If a user matches a rule but the user cannot authenticate in the other domain because it is untrusted, the mapping fails.

The SVM automatically discovers all bidirectional trusted domains, which are used for multi-domain user mapping searches. Alternatively, you can configure a list of preferred trusted domains that are used for name mapping searches instead of the list of automatically discovered trusted domains.

Regular expressions are not case-sensitive when mapping from Windows to UNIX. However, they are case-sensitive for Kerberos-to-UNIX and UNIX-to-Windows mappings.

As an example, the following rule converts the Windows user named “jones” in the domain named “ENG” into the UNIX user named “jones”.

Pattern	Replacement
ENG\\jones	jones

Note that the backslash is a special character in regular expressions and must be escaped with another backslash.

The caret (^), underscore (_), and ampersand (&) characters can be used as prefixes for digits in replacement patterns. These characters specify uppercase, lowercase, and initial-case transformations, respectively. For instance:

- If the initial pattern is (.+) and the replacement pattern is \1, then the string jOe is mapped to jOe (no change).
- If the initial pattern is (.+) and the replacement pattern is _1, then the string jOe is mapped to joe.
- If the initial pattern is (.+) and the replacement pattern is \^1, then the string jOe is mapped to JOE.
- If the initial pattern is (.+) and the replacement pattern is \&1, then the string jOe is mapped to Joe.

If the character following a backslash-underscore (_), backslash-caret (\^), or backslash-ampersand (\&) sequence is not a digit, then the character following the backslash is used verbatim.

The following example converts any Windows user in the domain named “ENG” into a UNIX user with the same name in NIS.

Pattern	Replacement
ENG\\(.+)	\1

The double backslash (\\) matches a single backslash. The parentheses denote a subexpression but do not match any characters themselves. The period matches any single character. The asterisk matches zero or more of the previous expression. In this example, you are matching ENG\ followed by one or more of any character. In the replacement, \1 refers to whatever the first subexpression matched. Assuming the Windows user ENG\jones, the replacement evaluates to jones; that is, the portion of the name following ENG\.

Note: If you are using the CLI, you must delimit all regular expressions with double quotation marks ("). For instance, to enter the regular expression (.+) in the CLI, type "(.+)" at the command prompt. Quotation marks are not required in the Web UI.

For further information about regular expressions, see your UNIX system administration documentation, the online UNIX documentation for `sed` or `regex`, or *Mastering Regular Expressions*, published by O'Reilly and Associates.

Creating 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 Storage Virtual Machine (SVM), Data ONTAP supports up to 1,024 name mappings for each direction.

Step

1. To create a name mapping, enter the following command:

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

   `-vserver` *`vserver_name`* specifies the SVM name.

   `-direction` {`krb-unix`|`win-unix`|`unix-win`} specifies the mapping direction.

   `-position` *`integer`* specifies the desired position in the priority list of a new mapping.

   `-pattern` *`text`* specifies the pattern to be matched, up to 256 characters in length.

   `-replacement` *`text`* specifies the replacement pattern, up to 256 characters in length.

When Windows-to-UNIX mappings are created, any CIFS clients that have open connections to the Data 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\John.

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

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. The mapping maps every CIFS user in the domain ENG to users in the NIS domain associated with the SVM.

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

Configuring 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***

Configuring local UNIX users and groups

You can use local UNIX users and groups for authentication and name mappings.

Creating a local UNIX user

You can use the `vserver services unix-user create` command to create local UNIX users. A local UNIX user is a UNIX user you create on the Storage Virtual Machine (SVM) as a UNIX name services option and to be used in the processing of name mappings.

Step

1. To create a local UNIX user, enter the following command:
   `vserver services unix-user create -vserver` *`vserver_name`* `-user` *`user_name`* `-id` *`integer`* `-primary-gid` *`integer`* `-full-name` *`full_name`*

   `-vserver` *`vserver_name`* specifies the SVM name.

   `-user` *`user_name`* specifies the user name.

   `-id` *`integer`* specifies the user ID.

   `-primary-gid` *`integer`* specifies the primary group ID.

   `-full-name` *`full_name`* specifies the full name of the user.

Example

The following command creates a local UNIX user named johnm on the SVM named vs1. The user has the ID 123 and the primary group ID 100. The user's full name is "John Miller".

```
node::> vserver services unix-user create -vserver vs1 -user johnm -
id 123
-primary-gid 100 -full-name "John Miller"
```

Loading local UNIX users from a URI

You can use the `vserver services unix-user load-from-uri` command to load one or more local UNIX users into Storage Virtual Machines (SVMs) from a uniform resource identifier (URI).

About this task

The URI must contain user information in the UNIX `/etc/passwd` format:
`user_name``:` *`password`*`:` *`user_ID`*`:` *`group_ID`*`:` *`full_name`*

The command discards the value of the *`password`* field and of the fields after the *`full_name`* field (*`home_directory`* and *`shell`*).

Step

1. To load one or more local UNIX users into SVMs from a URI, enter the following command:

   vserver services unix-user load-from-uri -vserver *vserver_name* -uri {ftp|http}://*uri* -overwrite {true|false}

   `-vserver` *vserver_name* specifies the SVM name.

   `-uri` {`ftp`|`http`}://*uri* specifies the URI to load from.

   `-overwrite` {`true`|`false`} specifies whether to overwrite entries. The default is `false`.

Example

The following command loads user information from the URI `ftp://ftp.example.com/passwd` into the SVM named vs1. Existing users on the SVM are not overwritten by information from the URI.

```
node::> vserver services unix-user load-from-uri -vserver vs1
-uri ftp://ftp.example.com/passwd -overwrite false
```

Creating a local UNIX group

You can use the `vserver services unix-group create` command to create UNIX groups that are local to the Storage Virtual Machine (SVM). Local UNIX groups are used with local UNIX users.

Step

1. To create a local UNIX group, enter the following command:

   vserver services unix-group create -vserver *vserver_name* -name *group_name* -id *integer*

   `-vserver` *vserver_name* specifies the SVM name.

   `-name` *group_name* specifies the group name.

   `-id` *integer* specifies the group ID.

Example

The following command creates a local group named eng on the SVM named vs1. The group has the ID 101.

```
vs1::> vserver services unix-group create -vserver vs1 -name eng -
id 101
```

Loading local UNIX groups from a URI

You can use the `vserver services unix-group load-from-uri` command to load one or more local UNIX groups into Storage Virtual Machines (SVMs) from a uniform resource identifier (URI).

About this task

The URI must contain user information in the UNIX `/etc/group` format:
`group_name``:` *`password`*`:` *`group_ID`*`:` *`comma_separated_list_of_users`*

The command discards the value of the *`password`* field.

Step

1. To load one or more local UNIX groups into SVMs from URI, enter the following command:

   `vserver services unix-group load-from-uri -vserver` *`vserver_name`* `-uri {ftp|http}://`*`uri`* `-overwrite {true|false}`

   `-vserver` *`vserver_name`* specifies the SVM name.

   `-uri` {`ftp`|`http`}://*`uri`* specifies the URI to load from.

   `-overwrite` {`true`|`false`} specifies whether to overwrite entries. The default is `false`.

Example

The following command loads group information from the URI `ftp://ftp.example.com/group` into the SVM named vs1. Existing groups on the SVM are not overwritten by information from the URI.

```
vs1::> vserver services unix-group load-from-uri -vserver vs1
-uri ftp://ftp.example.com/group -overwrite false
```

Adding a user to a local UNIX group

You can use the `vserver services unix-group adduser` command to add a user to a UNIX group that is local to the Storage Virtual Machine (SVM).

Step

1. To add a user to a local UNIX group, enter the following command:

   `vserver services unix-group adduser -vserver` *`vserver_name`* `-name` *`group_name`* `-username` *`user_name`*

   `-vserver` *`vserver_name`* specifies the SVM name.

`-name` *`group_name`* specifies the name of the UNIX group to add the user to.

`-username` *`user_name`* specifies the user name of the user to add to the group.

Example

The following command adds a user named max to a local UNIX group named eng on the SVM named vs1.

```
vs1::> vserver services unix-group adduser -vserver vs1 -name eng
-username max
```

Loading netgroups into SVMs

You can use the `vserver services netgroup load` command to load netgroups into Storage Virtual Machines (SVMs) from a uniform resource identifier (URI).

Step

1. To load netgroups into SVMs from an FTP or HTTP URI, enter the following command:

   `vserver services netgroup load -vserver` *`vserver_name`* `-source {ftp|http}://`*`uri`*

   `-vserver` *`vserver_name`* specifies the SVM name.

   `-source` {`ftp`|`http`}://*`uri`* specifies the URI to load from.

   The file must use the same proper netgroup text file format that is used to populate NIS.

Example

The following command loads netgroup definitions into the SVM named vs1 from the HTTP URL `http://intranet/downloads/corp-netgroup`:

```
vs1::> vserver services netgroup load -vserver vs1
-source http://intranet/downloads/corp-netgroup
```

Creating a NIS domain configuration

If you specified NIS as a name service option during Storage Virtual Machine (SVM) setup, you must create a NIS domain configuration for the SVM. You can use the `vserver services nis-domain create` command to create a NIS domain configuration.

About this task

You can create multiple NIS domains. However, you can only use one that is set to `active`.

Step

1. Use the `vserver services nis-domain create` command to create a NIS domain configuration.

Example

The following command creates a NIS domain configuration for a NIS domain called nisdomain on the SVM named vs1 with a NIS server at IP address 192.0.2.180 and makes it active.

```
vs1::> vserver services nis-domain create -vserver vs1 -domain
nisdomain -active true -servers 192.0.2.180
```

Support for NFS over IPv6

Starting with Data ONTAP 8.2, NFS clients can access files on your storage system over an IPv6 network.

Enabling IPv6 for NFS

If you want NFS clients to be able to access data on the storage system over an IPv6 network, Data ONTAP requires several configuration changes.

Steps

1. Enable IPv6 on the cluster.

   This step must be performed by a cluster administrator. For more information about enabling IPv6 on the cluster, see the *Clustered Data ONTAP Network Management Guide*.

2. Configure data LIFs for IPv6.

This step must be performed by a cluster administrator. For more information about configuring LIFs, see the *Clustered Data ONTAP Network Management Guide*.

3. Configure NFS export policies and rules for NFS client access.

   If you want to match clients by IPv6 address, be sure to configure the export rules accordingly.

Where to find information about setting up file access to Infinite Volumes

For more information about how to set up NFS file access to Infinite Volumes, see the *Clustered Data ONTAP Infinite Volumes Management Guide*.

Managing file access using NFS

After you have enabled NFS on the Storage Virtual Machine (SVM) and configured it, there are a number of tasks you might want to perform to manage file access using NFS.

Controlling 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**

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, Data ONTAP does not allow NFS clients that use nonreserved ports, meaning ports with numbers higher than 1,023, to mount NFS exports.

Commands for managing NFS servers

There are specific Data 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`

See the man page for each command for more information.

Commands for managing name mappings

There are specific Data 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	`vserver name-mapping swap`
Modify a name mapping	`vserver name-mapping modify`
Delete a name mapping	`vserver name-mapping delete`

See the man page for each command for more information.

Commands for managing local UNIX users

There are specific Data ONTAP commands for managing local UNIX users.

If you want to...	Use this command...
Create a local UNIX user	`vserver services unix-user create`
Display local UNIX users	`vserver services unix-user show`
Modify a local UNIX user	`vserver services unix-user modify`
Delete a local UNIX user	`vserver services unix-user delete`

See the man page for each command for more information.

Commands for managing local UNIX groups

There are specific Data ONTAP commands for managing local UNIX groups.

If you want to...	Use this command...
Add a user to a local UNIX group	`vserver services unix-group adduser`
Create a local UNIX group	`vserver services unix-group create`
Display local UNIX groups	`vserver services unix-group show`
Modify a local UNIX group	`vserver services unix-group modify`
Delete a user from a local UNIX group	`vserver services unix-group deluser`
Delete a local UNIX group	`vserver services unix-group delete`

See the man page for each command for more information.

Verifying the status of netgroup definitions

After loading netgroups into the Storage Virtual Machine (SVM), you can use the `vserver services netgroup status` command to verify the status of netgroup definitions. This enables you to determine whether netgroup definitions are consistent on all of the nodes that back the SVM.

Steps

1. Set the privilege level to advanced:
   `set -privilege advanced`

2. Verify the status of netgroup definitions:

   `vserver services netgroup status`

   The command displays the following information:

   - SVM name
   - Node name
   - Load time for netgroup definitions
   - Hash value of the netgroup definitions

   You can display additional information in a more detailed view. See the reference page for the command for details.

3. Return to the admin privilege level:

   `set -privilege admin`

Example

The following command displays netgroup status for all SVMs.

```
vs1::> set -privilege advanced

Warning: These advanced commands are potentially dangerous; use them only when
         directed to do so by technical support.
Do you wish to continue? (y or n): y

vs1::*> vserver services netgroup status
Virtual
Server    Node            Load Time           Hash Value
--------- --------------- ------------------- --------------------------------
vs1
          node1           9/20/2006 16:04:53  e6cb38ec1396a280c0d2b77e3a84eda2
          node2           9/20/2006 16:06:26  e6cb38ec1396a280c0d2b77e3a84eda2
          node3           9/20/2006 16:08:08  e6cb38ec1396a280c0d2b77e3a84eda2
          node4           9/20/2006 16:11:33  e6cb38ec1396a280c0d2b77e3a84eda2
```

Commands for managing NIS domain configurations

There are specific Data ONTAP commands for managing NIS domain configurations.

If you want to...	Use this command...
Create a NIS domain configuration	`vserver services nis-domain create`
Display NIS domain configurations	`vserver services nis-domain show`
Modify a NIS domain configuration	`vserver services nis-domain modify`
Delete a NIS domain configuration	`vserver services nis-domain delete`

See the man page for each command for more information.

Commands for managing LDAP client configurations

There are specific Data ONTAP commands for managing LDAP client configurations.

If you want to...	Use this command...
Create an LDAP client configuration	`vserver services ldap client create`
Display LDAP client configurations	`vserver services ldap client show`
Modify an LDAP client configuration	`vserver services ldap client modify`
Delete an LDAP client configuration	`vserver services ldap client delete`

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

See the man page for each command for more information.

Commands for managing LDAP configurations

There are specific Data ONTAP commands for managing LDAP configurations.

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

See the man page for each command for more information.

Commands for managing LDAP client schema templates

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

Note: The `vserver services ldap client schema copy`, `modify`, and `delete` commands are only available at the advanced privilege level and higher.

If you want to...	Use this command...
Copy an existing LDAP schema template	`vserver services ldap client schema copy`
Display LDAP schema templates	`vserver services ldap client schema show`
Modify an LDAP schema template	`vserver services ldap client schema modify`
Delete an LDAP schema template	`vserver services ldap client schema delete`

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

See the man page for each command for more information.

How the access cache works

The Data ONTAP access cache reduces the likelihood of having to perform a reverse DNS lookup or parse netgroups when granting or denying an NFS client access to a volume or qtree. This results in performance improvements due to less time used for DNS lookups.

Whenever an NFS client attempts to access a volume or qtree, Data ONTAP must determine whether to grant or deny access. Except in the most simple cases (for example, when a volume or qtree is exported with just the `ro` or `rw` option), Data ONTAP grants or denies access according to a value in the access cache that corresponds to the following things:

- The volume or qtree
- The NFS client's IP address, access type, and security type

This value might not exist in the access cache entry if Data ONTAP has not made a previous access determination for this particular NFS client volume or qtree combination. In this case, Data ONTAP grants or denies access according to the result of a comparison between the following things:

- The NFS client’s IP address (or host name, if necessary), access type, and security type
- The volume or qtree export rules

Data ONTAP then stores the result of this comparison in the access cache for five minutes.

Displaying information about NFS Kerberos configurations

You can use the `vserver nfs kerberos-config show` command to display information about NFS Kerberos configurations. This enables you to determine how NFS Kerberos is configured.

Step

1. To display information about NFS Kerberos configurations, enter the following command:

   `vserver nfs kerberos-config show`

   The command displays the following information:

   - Virtual server name
   - Logical interface name
   - Logical interface IP address
   - Whether Kerberos is enabled or disabled
   - Kerberos SPN
   - Numeric ID of the configuration

   You can display additional information in a more detailed view. See the reference page for the command for details.

Example

The following command displays detailed information about an NFS Kerberos configuration:

```
vs1::> vserver nfs kerberos-config show -vserver vs1
-lif datalif1

        Virtual Server: vs1
     Logical Interface: datalif1
            Ip Address: 172.19.4.1
      Kerberos Enabled: Disabled
Service Principal Name: nfs/security.example.com@ENG.EXAMPLE.COM
```

Modifying an NFS Kerberos configuration

You can use the `vserver nfs kerberos-config modify` command to modify a Kerberos configuration for NFS. This enables you to enable or disable the NFS-enabled Storage Virtual Machine (SVM) to use Kerberos authentication.

Step

1. Use the `vserver nfs kerberos-config modify` command to modify an NFS Kerberos configuration

Examples

The following command enables an NFS Kerberos configuration on the SVM named vs1 and a logical interface named datalif1. The SPN is nfs/security.example.com@eng.example.com and the keytab file to be loaded is at the URL ftp://ftp.example.com/keytab.

```
vs1::> vserver nfs kerberos-config modify -vserver vs1 -lif
datalif1 -kerberos enable -spn nfs/
security.example.com@ENG.EXAMPLE.COM -admin-username admin -keytab-
uri ftp://ftp.example.com/keytab
```

Data ONTAP then prompts the user for the password for the admin-user. The admin-user should have permission on the KDC to add the principal to the principal's database.

The following command disables the NFS Kerberos configuration that was created in the previous example.

```
vs1::> vserver nfs kerberos-config modify -vserver vs1 -lif
datalif1 -kerberos disable
```

Commands for managing Kerberos realm configurations

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

If you want to...	Use this command...
Create a Kerberos realm configuration	`vserver services kerberos-realm create`
Display Kerberos realm configurations	`vserver services kerberos-realm show`

If you want to...	Use this command...
Modify Kerberos realm configurations	`vserver services kerberos-realm modify`
Delete a Kerberos realm configuration	`vserver services kerberos-realm delete`

See the man page for each command for more information.

Commands for managing export policies

There are specific Data 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 Data 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`

See the man page for each command for more information.

Managing file locks

You can display information about the current locks for a Storage Virtual Machine (SVM) as a first step to determining why a client cannot access a volume or file. You can use this information if you need to break file locks.

For information about how file locks affect Infinite Volumes, see the *Clustered Data ONTAP Infinite Volumes Management Guide*.

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 Data 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.

How Data ONTAP treats read-only bits

The read-only bit is a binary digit, which holds a value of 0 or 1, that is set on a file-by-file basis to reflect whether a file is writable (disabled) or read-only (enabled).

SMB clients that use MS-DOS and 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.

Data ONTAP can set a read-only bit on a file when an SMB client that uses MS-DOS or Windows creates that file. Data 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 Data 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, Data ONTAP enables the read-only bit for that file.

- If an NFS client enables any write permission bit, Data 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, Data 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, Data ONTAP enables the owner's write permission bit for the file.
- Files with the read-only bit enabled are writable only by root.

Note: 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.

Displaying 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. If you do not specify any parameter, the command displays the following information:

- SVM name
- Volume name of the FlexVol volume or the name of the namespace constituent for the Infinite Volume
- Path of the locked object
- Logical interface name
- Protocol by which the lock was established
- Type of lock
- Client

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 of these lock types. 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`***

`-vserver` *`vserver_name`* specifies the SVM name.

`-volume` *`volume_name`* specifies the volume name of the FlexVol volume or the name of the namespace constituent for the Infinite Volume.

`-path` *`path`* specifies the path.

`-lif` *`lif`* specifies the logical interface.

`-lockid` specifies the universally unique identifier for the lock.

4. Return to the admin privilege level:

   ```
   set -privilege admin
   ```

Modifying 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`	ibm.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
   ```

Managing NFSv4 ACLs

You can enable, disable, set, modify, and view NFSv4 access control lists (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.

For access checking, CIFS users are mapped to UNIX users. The mapped UNIX user and that user's group membership are checked against the ACL.

If a file or directory has an ACL, that ACL is used to control access no matter what protocol—NFSv3, NFSv4, or CIFS—is used to access the file or directory and is used even if NFSv4 is no longer enabled on the system.

Files and directories inherit ACEs from NFSv4 ACLs on parent directories (possibly with appropriate modifications) as long as the ACEs have been tagged with the appropriate inheritance flags.

The maximum number of ACEs for each ACL is 1,024, as defined by the `-v4-acl-max-aces` parameter. The default for this parameter is 400. When the `-v4-acl-preserve` parameter is enabled and an object has both UNIX mode bits and ACLs, the ACL requires three mandatory ACEs to grant the proper permissions to OWNER, GROUP, and EVERYONE. These mandatory ACEs are applied automatically by Data ONTAP when a user performs a `chmod` operation to set mode bits. If you set an ACL that is missing any of the mandatory ACEs, Data ONTAP limits the number of ACEs for that ACL to reserve space for any missing mandatory ACEs to ensure that they can be added later should you decide to set mode bits as well.

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.

  Note: 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.

Enabling or disabling modification of NFSv4 ACLs

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

To determine whether it can delete a file, Data 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.

Enabling or disabling 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.

NFSv4.0 ACLs are not supported for Storage Virtual Machines (SVMs) with Infinite Volume.

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**

Modifying 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 Data 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 by entering the following command:

   `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`

Managing NFSv4 file delegations

You can enable and disable NFSv4 file delegations and retrieve NFSv4 file delegation statistics.

How NFSv4 file delegations work

Data ONTAP supports read and write file delegations in accordance with RFC 3530.

As specified in RFC 3530, when an NFSv4 client opens a file, Data ONTAP can delegate further handling of opening and writing requests to the opening client. There are two types of file delegations: read and write. A read file delegation allows a client to handle requests to open a file for reading that do not deny read access to others. A write file delegation allows the client to handle all open requests.

Delegation works on files within any style of qtree, whether or not opportunistic locks (oplocks) have been enabled.

Delegation of file operations to a client can be recalled when the lease expires, or when the storage system receives the following requests from another client:

- Write to file, open file for writing, or open file for “deny read”
- Change file attributes
- Rename file
- Delete file

When a lease expires, the delegation state is revoked and all of the associated states are marked “soft”. This means that if the storage system receives a conflicting lock request for this same file from another client before the lease has been renewed by the client previously holding the delegation,

the conflicting lock is granted. If there is no conflicting lock and the client holding the delegation renews the lease, the soft locks are changed to hard locks and are not removed in the case of a conflicting access. However, the delegation is not granted again upon a lease renewal.

When the server reboots, the delegation state is lost. Clients can reclaim the delegation state upon reconnection instead of going through the entire delegation request process again. When a client holding a read delegation reboots, all delegation state information is flushed from the storage system cache upon reconnection. The client must issue a delegation request to establish a new delegation.

Note: NFSv4 File delegations are not supported on Storage Virtual Machines (SVMs) with Infinite Volume.

Enabling or disabling 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.

NFSv4 file delegations are not supported on Storage Virtual Machines (SVMs) with Infinite Volume.

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.

Enabling or disabling 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.

NFSv4 file delegations are not supported on Storage Virtual Machines (SVMs) with Infinite Volume.

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.

Configuring NFSv4 file and record locking

You can configure NFSv4 file and record locking by specifying the locking lease period and grace period.

About NFSv4 file and record locking

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

In accordance with RFC 3530, Data ONTAP "defines a single lease period for all state held by an NFS client. If the client does not renew its lease within the defined period, all states associated with the client's lease may be released by the server." The client can renew its lease explicitly or implicitly by performing an operation, such as reading a file.

Furthermore, Data ONTAP defines a grace period, which is a period of special processing in which clients attempt to reclaim their locking state during a server recovery.

Term	Definition (see RFC 3530)
Lease	The time period in which Data ONTAP irrevocably grants a lock to a client.
Grace period	The time period in which clients attempt to reclaim their locking state from Data ONTAP during server recovery.
Lock	Refers to both record (byte-range) locks as well as file (share) locks unless specifically stated otherwise.

Specifying the NFSv4 locking lease period

To specify the NFSv4 locking lease period (that is, the time period in which Data 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
   ```

Specifying 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 Data 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, Data 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 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 gives 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 allowing the clients faster access to the data and avoiding extra cluster communication.

Support for NFSv4 referrals is not uniformly available in all NFSv4 clients. In an environment where not all clients support this feature, you should not enable referrals. If the feature is enabled and a client that does not support it receives a referral from the server, the client cannot access the volume and experiences failures.

See RFC3530 for details about referrals.

Note: Referrals are not supported on SVMs with Infinite Volume.

Enabling or disabling NFSv4 referrals

You can enable NFSv4 referrals on Storage Virtual Machines (SVMs) with FlexVol volumes 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.

Before you begin

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`

Displaying 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.

   #### Example

   statistics catalog object show -object nfs*

   For more information about the `statistics` commands, see the *Clustered Data ONTAP System Administration Guide for Cluster Administrators*.

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
```

Support for VMware vStorage over NFS

Data 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:

- vStorage is not supported with FlexCache.
- vStorage is not supported on Storage Virtual Machines (SVMs) with Infinite Volume.
- 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

Enabling or disabling VMware vStorage over NFS

You can enable or disable support for VMware vStorage over NFS on Storage Virtual Machines (SVMs) with FlexVol volumes 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 by entering the following command:

   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 IBM N Series NFS Plug-in for VMware VAAI*.

Enabling or disabling rquota support

Data ONTAP supports the remote quota protocol version 1 (rquota v1). The rquota protocol enables NFS clients to obtain quota information for users and groups from a remote machine. You can enable rquota on Storage Virtual Machines (SVMs) with FlexVol volumes 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 the *Clustered Data ONTAP Logical Storage Management Guide*.

NFSv3 performance improvement by modifying the TCP maximum read and write size

You might be able to improve the performance of NFSv3 clients connecting to storage systems over a high-latency network by modifying the TCP maximum read and write 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 read and write 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 due to 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 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. If you configure the storage system to support 1 MB maximum read size but the client only supports 64 KB, then the mount read 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.

Modifying the NFSv3 TCP maximum read and write size

You can modify the `-v3-tcp-max-read-size` and `-v3-tcp-max-write-size` options to change the NFSv3 TCP maximum read and write size. Modifying these options can help improve NFSv3 performance over TCP in some storage environments.

Before you begin

All nodes in the cluster must be running Data ONTAP 8.1 or later.

About this task

You can modify these options individually for each 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 to...	Enter the command...
Modify the NFSv3 TCP maximum read size	**`vserver nfs modify -vserver *vserver_name* -v3-tcp-max-read-size *integer_max_read_size*`**
Modify the NFSv3 TCP maximum write size	**`vserver nfs modify -vserver *vserver_name* -v3-tcp-max-write-size *integer_max_write_size*`**

Option	Range	Default
`-v3-tcp-max-read-size`	8192 to 1048576 bytes	65536 bytes
`-v3-tcp-max-write-size`	8192 to 65536 bytes	65536 bytes

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

3. Return to the admin privilege level:

   `set -privilege admin`

4. Use the `vserver nfs show` command to verify the changes.

5. If any clients use static mounts, unmount and remount for the new parameter size to take effect.

Example

The following command sets the NFSv3 TCP maximum read size to 1048576 bytes on the SVM named vs1:

```
vs1::> vserver nfs modify -vserver vs1 -v3-tcp-max-read-size 1048576
```

Auditing NAS file access events on SVMs with FlexVol volumes

Auditing for NAS file access events is a security measure that enables you to track and log SMB and NFS file and folder access events on objects stored on Storage Virtual Machines (SVMs) with FlexVol volumes. This helps you track potential security problems and provides evidence of any file access security breaches.

How auditing works

Before you plan and configure your auditing configuration, you should understand how auditing works.

Basic auditing concepts

To understand auditing in Data ONTAP, you should be aware of some basic auditing concepts.

Staging files	The intermediate binary files on individual nodes where audit records are stored prior to consolidation and conversion. Staging files are contained in staging volumes.
Staging volume	A dedicated volume created by Data ONTAP to store staging files. There is one staging volume per aggregate. Staging volumes are shared by all audit-enabled Storage Virtual Machines (SVMs) with volumes in that particular aggregate. Staging volumes are a type of system volume. System volumes are FlexVol volumes that contain special metadata, such as metadata for file services audit logs. System volumes are owned by the admin SVM and are visible across the cluster; therefore, there is no multi-tenancy for staging volumes. Only cluster administrators can view staging volumes. Additionally, cluster administrators can modify, or delete staging volumes, but they cannot create staging volumes.
Consolidation task	A task that takes the audit records from staging files across the member nodes of the SVM on a per-SVM basis and merges them in sorted chronological order and then converts them to a user-readable event log format specified in the auditing configuration — either the EVTX or XML file format. The converted event logs are stored in the audit event log directory that is specified in the SVM auditing configuration.

How the Data ONTAP auditing process works

The Data ONTAP auditing process is different than the Microsoft auditing process. Before you configure auditing, you should understand how the Data ONTAP auditing process works.

Audit records are initially stored in binary staging files on individual nodes. If auditing is enabled on an SVM, every member node maintains staging files for that SVM. Periodically, they are consolidated and converted to user-readable event logs, which are stored in the audit event log directory for the SVM.

Process when auditing is enabled on an SVM

Auditing can only be enabled on SVMs with FlexVol volumes. When the storage administrator enables auditing on the SVM, the auditing subsystem checks to determine if staging volumes are present. A staging volume must exist for each aggregate containing data volumes owned by the SVM. The auditing subsystem creates any needed staging volumes if they do not exist.

The auditing subsystem also completes other prerequisite tasks before auditing is enabled:

- The auditing subsystem verifies that the log directory path is available and does not contain symlinks.

  The log directory must already exist. The auditing subsystem does not assign a default log file location. If the log directory path specified in the auditing configuration is not a valid path, auditing configuration creation fails with the following error:
  `The specified path "/<path>" does not exist in the namespace belonging to Vserver "<Vserver_name>"`

  Configuration creation fails if the directory exists but contains symlinks.
- Auditing schedules the consolidation task.

After these tasks are completed successfully, auditing is enabled. The SVM auditing configuration and the log files persist across a reboot or if the NFS or CIFS servers are stopped or restarted.

Event log consolidation

Log consolidation is a scheduled task that runs on a routine basis until auditing is disabled. When auditing is disabled, the final run of the consolidation task ensures that all the remaining logs are consolidated.

Guaranteed auditing

By default, auditing is guaranteed. Data ONTAP guarantees that all auditable file access events (as specified by configured audit policy ACLs) are recorded, even if a node is unavailable. A requested file operation cannot complete until the audit record for that operation is saved to the staging volume on persistent storage. If audit records cannot be committed to the disk in the staging files, either because of insufficient space or because of other issues, client operations are denied.

Consolidation process when a node is unavailable

If a node containing volumes belonging to an SVM with auditing enabled is unavailable, the behavior of the auditing consolidation task depends on whether the node's SFO partner (or the HA partner in the case of a two-node cluster) is available.

- If the staging volume is available through the SFO partner, the staging volumes last reported from the node are scanned, and consolidation proceeds normally.
- If the SFO partner is not available, the task creates a partial log file.
  When a node is not reachable, the consolidation task consolidates the audit records from the other available nodes of that SVM. But to identify that it is not complete, the task adds the suffix `.partial` to the consolidated file name.
- After the unavailable node is available, the audit records in that node are consolidated with the audit records from the other nodes at that point of time.
- All audit records are preserved.

Event log rotation

Audit event log files are rotated when they reach a configured threshold log size or on a configured schedule. When an event log file is rotated, the scheduled consolidation task first renames the active converted file to a time-stamped archive file and creates a new active converted event log file.

Process when auditing is disabled on the SVM

When auditing is disabled on the SVM, the consolidation task is triggered one final time. All outstanding, recorded audit records are logged in user-readable format. Existing event logs stored in the event log directory are not deleted when auditing is disabled on the SVM and are available for viewing.

After all existing staging files for that SVM are consolidated, the consolidation task is removed from the schedule. Disabling the auditing configuration for the SVM does not remove the auditing configuration. A storage administrator can reenable auditing at any time.

Aggregate space considerations when enabling auditing

When an auditing configuration is created and auditing is enabled on at least one Storage Virtual Machine (SVM) in the cluster, the auditing subsystem creates staging volumes on all existing aggregates and on all new aggregates that are created. You need to be aware of certain aggregate space considerations when you enable auditing on the cluster.

Staging volume creation might fail due to non-availability of space in an aggregate. This might happen if you create an auditing configuration and existing aggregates do not have enough space to contain the staging volume.

You should ensure that there is enough space on existing aggregates for the staging volumes before enabling auditing on an SVM.

Auditing requirements and considerations

Before you configure and enable auditing on your Storage Virtual Machine (SVM) with FlexVol volumes, you need to be aware of certain requirements and considerations.

- Before you can enable auditing on your SVM, all nodes in the cluster must be running Data ONTAP 8.2 or later.
- The maximum number of audit-enabled SVMs supported in a cluster is 50.
- Auditing is not tied to CIFS or NFS licensing.

  You can configure and enable auditing even if CIFS and NFS licenses are not installed on the cluster.
- NFS auditing supports security ACEs (type U).
- For NFS auditing, there is no mapping between mode bits and audit ACEs.

  When converting ACLs to mode bits, audit ACEs are skipped. When converting mode bits to ACLs, audit ACEs are not generated.
- The directory specified in the auditing configuration must exist.

  If it does not exist, the command to create the auditing configuration fails.
- The directory specified in the auditing configuration must meet the following requirements:
  - The directory must not contain symbolic links.

    If the directory specified in the auditing configuration contains symbolic links, the command to create the auditing configuration fails.
  - You must specify the directory by using an absolute path.

    You should not specify a relative path, for example, `/vs1/../`.
- Auditing is dependent on having available space in the staging volumes.

  You must be aware of and have a plan for ensuring that there is sufficient space for the staging volumes in aggregates that contain audited volumes.
- Auditing is dependent on having available space in the volume containing the directory where converted audit event logs are stored.

  You must be aware of and have a plan for ensuring that there is sufficient space in the volumes used to store event logs. You can specify the number of audit logs to retain in the auditing directory by using the `-rotate-limit` parameter when creating an auditing configuration, which can help to ensure that there is enough available space for the audit logs in the volume.

What the supported audit event log formats are

Supported file formats for the converted audit event logs are `EVTX` and `XML` file formats.

You can specify the type of file format when you create the auditing configuration. By default, Data ONTAP converts the binary logs to the `EVTX` file format.

Viewing audit event logs

You can use audit event logs to determine whether you have adequate file security and whether there have been improper file and folder access attempts. You can view and process audit event logs saved in the EVTX or XML file formats.

- EVTX file format

  You can open the converted EVTX audit event logs as saved files using Microsoft Event Viewer. There are two options that you can use when viewing event logs using Event Viewer:

  - General view

    Information that is common to all events is displayed for the event record. In this version of Data ONTAP, the event-specific data for the event record is not displayed. You can use the detailed view to display event-specific data.
  - Detailed view

    A friendly view and a XML view are available. The friendly view and the XML view display both the information that is common to all events and the event-specific data for the event record.
- XML file format

  You can view and process XML audit event logs on third-party applications that support the XML file format. XML viewing tools can be used to view the audit logs provided you have the XML schema and information about definitions for the XML fields. For more information about obtaining the XML schema and documents related to XML definitions, contact technical support or your account team.

SMB file and folder access events that can be audited

Data ONTAP can audit certain SMB file and folder access events. Knowing what access events can be audited is helpful when interpreting results from the converted audit event logs.

The following SMB file and folder access events can be audited:

Event ID (EVT/EVTX)	Event	Description	Category
560/4656	Open Object/ Create Object	OBJECT ACCESS: Object (file or directory) open.	File Access

Event ID (EVT/EVTX)	Event	Description	Category
567/4663	Read Object/ Write Object/Get Object Attributes/Set Object Attributes	OBJECT ACCESS: Object access attempt (read, write, get attribute, set attribute). **Note:** For this Event, Data ONTAP audits only the first SMB read and first SMB write operation (success or failure) on an object. This prevents Data ONTAP from creating excessive log entries when a single client opens an object and performs many successive read or write operations to the same object.	File Access
N/A/4664	Hard link	OBJECT ACCESS: An attempt was made to create a hard link.	File Access
N/A/N/A Data ONTAP Event ID 9999	Rename Object	OBJECT ACCESS: Object renamed. This is a Data ONTAP event. It is not currently supported by Windows as a single event.	File Access
N/A/N/A Data ONTAP Event ID 9998	Unlink Object	OBJECT ACCESS: Object unlinked. This is a Data ONTAP event. It is not currently supported by Windows as a single event.	File Access

Note: The object path printed in an audit record is the relative path from the root of the containing volume. For example, consider the following volume information:

```
                          Junction                       Junction
Vserver   Volume          Active   Junction Path         Path Source
--------- ------------ -------- ------------------ -----------
vs1       data         true     /data              RW_volume
vs1       data1        true     /data/data1        RW_volume
```

If a user accesses a file with the path `/data/data1/dir1/file.txt`, the path used in the `<ObjectName>` tag in the event contained in the audit logs is `/data1/dir1/file.txt`.

NFS file and directory access events that can be audited

Data ONTAP can audit certain NFS file and directory access events. Knowing what access events can be audited is helpful when interpreting results from the converted audit event logs.

You can audit the following NFS file and directory access events:

- READ
- OPEN

- CLOSE
- READDIR
- WRITE
- SETATTR
- CREATE
- LINK
- OPENATTR
- REMOVE
- GETATTR
- VERIFY
- NVERIFY
- RENAME

To reliably audit NFS RENAME events, you should set audit ACEs on directories instead of files because file permissions are not checked for a RENAME operation if the directory permissions are sufficient.

Planning the auditing configuration

Before you configure auditing on Storage Virtual Machines (SVMs) with FlexVol volumes, you must understand which configuration options are available and plan the values that you want to set for each option. This information can help you configure the auditing configuration that meets your business needs.

There are certain configuration parameters that are common to all auditing configurations.

Additionally, there are certain parameters that you can use to specify which of two methods are used when rotating the consolidated and converted audit logs. You can specify one of the two following methods when you configure auditing:

- Rotate logs based on log size
  This is the default method used to rotate logs.
- Rotate logs based on a schedule

Parameters common to all auditing configurations

There are two required parameters that you must specify when you create the auditing configuration. There are also two optional parameters that you can specify. The first optional parameter determines how many audit logs are retained in the audit log directory The second optional parameter specifies which log file format to use for the audit logs.

You can use the following list to determine what values to use for the parameters that are common to all auditing configurations:

Type of information	Option	Required	Include	Your values
SVM name Name of the SVM on which to create the auditing configuration. The SVM must already exist.	`-vserver` *vserver_name*	Yes	Yes	
Log destination path Specifies where the converted audit logs are stored. The path must already exist on the SVM. If the path is not valid, the audit configuration command fails.	`-destination` *text*	Yes	Yes	
Log file output format Determines the output format of the audit logs. The output format can be either Data ONTAP-specific `XML` or Microsoft Windows `EVTX` log format. By default, the output format is `EVTX`.	`-format {xml|evtx}`	No		
Log files rotation limit Determines how many audit log files to retain before rotating the oldest log file out. A value of `0` indicates that all the log files are retained. The default value is 0. For example, if you enter a value of 5, the last five log files are retained.	`-rotate-limit` *integer*	No		

Parameters used for determining when to rotate audit event logs

Rotate logs based on log size

The default is to rotate audit logs based on size. The default log size is 100 MB. If you want to use the default log rotation method and the default log size, you do not need to configure any specific parameters for log rotation. If you do not want to use the default log size, you can configure the `-rotate-size` parameter to specify a custom log size:

Type of information	Option	Required	Include	Your values
Log file size limit Determines the audit log file size limit.	`-rotate-size` {*integer*[KB|MB|GB|TB|PB]}	No		

Rotate logs based on a schedule

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

- If you configure time-based log rotation parameters, logs are rotated based on the configured schedule instead of log size.
- If you use time-based rotation, the `-rotate-schedule-minute` parameter is mandatory.
- All other time-based rotation parameters are optional.
- The rotation schedule is calculated by using all the time-related values.

  For example, if you specify only the `-rotate-schedule-minute` parameter, the audit log files are rotated based on the minutes specified on all days of the week, during all hours on all months of the year.
- If you specify only one or two time-based rotation parameters (for example, `-rotate-schedule-month` and `-rotate-schedule-minutes`), the log files are rotated based on the minute values that you specified on all days of the week, during all hours, but only during the specified months.

  For example, you can specify that the audit log is to be rotated during the months January, March, and August on all Mondays, Wednesdays, and Saturdays at 10:30 a.m.
- If you specify values for both `-rotate-schedule-dayofweek` and `-rotate-schedule-day`, they are considered independently.

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

You can use the following list of available auditing parameters to determine what values to use for configuring a schedule for audit event log rotations:

Type of information	Option	Required	Include	Your values
Log rotation schedule: Month Determines the monthly schedule for rotating audit logs. Valid values are `January` through `December`, and `all`. For example, you can specify that the audit log is to be rotated during the months January, March, and August.	`-rotate-schedule-month` *`chron_month`*	No		

Type of information	Option	Required	Include	Your values
Log rotation schedule: Day of week Determines the daily (day of week) schedule for rotating audit logs. Valid values are `January` through `December`, and `all`. For example, you can specify that the audit log is to be rotated on Tuesdays and Fridays, or during all the days of a week.	`-rotate-schedule-dayofweek` *chron_dayofweek*	No		
Log rotation schedule: Day Determines the day of the month schedule for rotating the audit log. Valid values range from `1` through `31`. For example, you can specify that the audit log is to be rotated on the 10th and 20th days of a month, or all days of a month.	`-rotate-schedule-day` *chron_dayofmonth*	No		
Log rotation schedule: Hour Determines the hourly schedule for rotating the audit log. Valid values range from `0` (midnight) to `23` (11:00 p.m.). Specifying `all` rotates the audit logs every hour. For example, you can specify that the audit log is to be rotated at 6 (6 a.m.) and 18 (6 p.m.).	`-rotate-schedule-hour` *chron_hour*	No		
Log rotation schedule: Minute Determines the minute schedule for rotating the audit log. Valid values range from `0` to `59`. For example, you can specify that the audit log is to be rotated at the 30th minute.	`-rotate-schedule-minute` *chron_minute*	Yes, if configuring schedule-based log rotation; otherwise, no.		

Creating a file and directory auditing configuration on SVMs

Creating a file and directory auditing configuration on your Storage Virtual Machine (SVM) with FlexVol volumes includes understanding the available configuration options, planning the configuration, and then configuring and enabling the configuration. You can then display information

about the auditing configuration to confirm that the resultant configuration is the desired configuration.

Steps

1. *Creating the auditing configuration* on page 120
   Before you can begin auditing file and directory events, you must create an auditing configuration on the Storage Virtual Machine (SVM).
2. *Enabling auditing on the SVM* on page 121
   After you finish setting up the auditing configuration, you must enable auditing on the Storage Virtual Machine (SVM).
3. *Verifying the auditing configuration* on page 122
   After completing the auditing configuration, you should verify that auditing is configured properly and is enabled.

Creating the auditing configuration

Before you can begin auditing file and directory events, you must create an auditing configuration on the Storage Virtual Machine (SVM).

Step

1. Using the information in the planning worksheet, create the auditing configuration by using the appropriate command:

If you want to create an auditing configuration that rotates audit logs based on...	Enter the command...
Log size	`vserver audit create -vserver vserver_name -destination path [-format {xml|evtx}] [-rotate-limit integer] [-rotate-size {integer[KB|MB|GB|TB|PB]}]`
A schedule	`vserver audit create -vserver vserver_name -destination path [-format {xml|evtx}] [-rotate-limit integer] [-rotate-schedule-month chron_month] [-rotate-schedule-dayofweek chron_dayofweek] [-rotate-schedule-day chron_dayofmonth] [-rotate-schedule-hour chron_hour] -rotate-schedule-minute chron_minute` **Note:** The `-rotate-schedule-minute` parameter is required if configuring time-based audit log rotation.

Examples

The following example creates an audit configuration for SVM vs1. The log format is `EVTX` (the default). The logs are stored in the `/audit_log` directory. The log file size limit is `200 MB`. The logs are rotated when they reach 200 MB in size:

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

The following example creates an audit configuration for SVM vs1 using size-based rotation. The log format is `EVTX` (the default). The log file size limit is `200 MB`, and the log rotation limit is `5`:

```
cluster1::> vserver audit create -vserver vs1 -destination /audit_log -
rotate-size 200MB -rotate-limit 5
```

The following example creates an audit configuration for SVM vs1 using time-based rotation. The log format is `EVTX` (the default). The audit logs are rotated monthly, at 12:30 p.m. on all days of the week:

```
cluster1::> vserver audit create -vserver vs1 -destination /audit_log -
rotate-size 200MB -rotate-schedule-month all -rotate-schedule-dayofweek all
-rotate-schedule-hour 12 -rotate-schedule-minute 30
```

Enabling auditing on the SVM

After you finish setting up the auditing configuration, you must enable auditing on the Storage Virtual Machine (SVM).

Before you begin

The SVM audit configuration must already exist.

Step

1. Enable auditing on the SVM:

   `vserver audit enable -vserver` ***`vserver_name`***

   Example

   `vserver audit enable -vserver vs1`

Verifying the auditing configuration

After completing the auditing configuration, you should verify that auditing is configured properly and is enabled.

Step

1. Verify the auditing configuration:

   vserver audit show -instance -vserver *vserver_name*

Example

The following example displays in list form all audit configuration information for Storage Virtual Machine (SVM) vs1. The EVTX-formatted logs are stored in the `/audit_log` directory. The log file size limit is `200 MB`, and the logs are rotated when they reach 200 MB in size. Auditing is enabled:

vserver audit show -instance -vserver vs1

```
                    Vserver: vs1
                    Auditing state: true
              Log Destination Path: /audit_log
                         Log Format: evtx
               Log File Size Limit: 200MB
      Log Rotation Schedule: Month: -
Log Rotation Schedule: Day of Week: -
       Log Rotation Schedule: Day: -
      Log Rotation Schedule: Hour: -
    Log Rotation Schedule: Minute: -
                Rotation Schedules: -
          Log Files Rotation Limit: 0
```

Configuring file and folder audit policies

Implementing auditing on file and folder access events is a two-step process. First you must create and enable an auditing configuration on Storage Virtual Machines (SVMs) with FlexVol volumes. Second, you must configure audit policies on the files and folders that you want to monitor. You can configure audit policies to monitor both successful and failed access attempts.

You can configure both SMB and NFS audit polices. SMB and NFS audit policies have different configuration requirements and audit capabilities.

If the appropriate audit policies are configured, Data ONTAP monitors SMB and NFS access events as specified in the audit policies only if the SMB or NFS servers are running.

Configuring audit policies on NTFS security-style files and directories

Before you can audit file and directory operations, you must configure audit policies on the files and directories for which you want to collect audit information. This is in addition to setting up and

enabling the audit configuration. You can configure NTFS audit policies by using the Windows Security tab or by using the Data ONTAP CLI.

Configuring NTFS audit policies using the Windows Security tab

You can configure audit policies on files and directories by using the **Windows Security** tab in the Windows Properties window. This is the same method used when configuring audit polices on data residing on a Windows client, which enables customers to use the same GUI interface that they are accustomed to using.

Before you begin

Auditing must be configured on the Storage Virtual Machine (SVM) that contains the data to which you are applying SACLs.

About this task

Configuring NTFS audit policies is done by adding entries to NTFS system access control lists (SACLs) that are associated with an NTFS security descriptor. The security descriptor is then applied to NTFS files and directories. These tasks are automatically handled by the Windows GUI. The security descriptor can contain discretionary access control lists (DACLs) for applying file and folder access permissions, system access control lists (SACLs) for file and folder auditing, or both SACLs and DACLs.

You can set NTFS audit policies for auditing access on individual files and folders using the Windows Security tab in the Windows Properties window by completing the following steps on a Windows host:

Steps

1. From the **Tools** menu in Windows Explorer, select **Map network drive**.
2. Complete the **Map Network Drive** box:
   a) Select a **Drive** letter.
   b) In the **Folder** box, type the CIFS server name that contains the share holding the data you would like to audit and the name of the share.

      #### Example

      If your CIFS server name is “CIFS_SERVER” and your share is named “share1”, you should enter `\\CIFS_SERVER\share1`.

      Note: You can specify the IP address of the data interface for the CIFS server instead of the CIFS server name.

   c) Click **Finish**.

   The drive you selected is mounted and ready with the Windows Explorer window displaying files and folders contained within the share.

3. Select the file or directory for which you want to enable auditing access.
4. Right-click on the file or directory, and select **Properties**.
5. Select the **Security** tab.
6. Click **Advanced**.
7. Select the **Auditing** tab.
8. Perform the desired actions:

If you want to....	Do the following
Set up auditing for a new user or group	**a.** Click **Add**. **b.** In the **Enter the object name to select** box, type the name of the user or grout that you want to add. **c.** Click **OK**.
Remove auditing from a user or group	**a.** In the **Enter the object name to select** box, select the user or group that you want to remove. **b.** Click **Remove**. **c.** Click **OK**. **d.** Skip the rest of this procedure.
Change auditing for a user or group	**a.** In the **Enter the object name to select** box, select the user or group that you want to change. **b.** Click **Edit**. **c.** Click **OK**.

If you are setting up auditing on a user or group or changing auditing on an existing user or group, the Auditing Entry for <object> box opens.

9. In the **Apply to** box, select how you want to apply this auditing entry.

   You can select one of the following:

   - **This folder, subfolders and files**
   - **This folder and subfolders**
   - **This folder only**
   - **This folder and files**
   - **Subfolders and files only**
   - **Subfolders only**
   - **Files only**

   If you are setting up auditing on a single file, the **Apply to** box is not active. The **Apply to** defaults to **This object only**.

Note: Since auditing takes SVM resources, select only the minimal level that provides the auditing events that meet your security requirements.

10. In the **Access** box, select what you want audited and whether you want to audit successful events, failure events or both.

    - To audit successful events, select the **Success** box.
    - To audit failure events, select the **Failure** box.

    You can audit the following events:

    - **Full control**
    - **Traverse folder / execute file**
    - **List folder / read data**
    - **Read attributes**
    - **Read extended attributes**
    - **Create files / write data**
    - **Create folders / append data**
    - **Write attributes**
    - **Write extended attributes**
    - **Delete subfolders and files**
    - **Delete**
    - **Read permissions**
    - **Change permissions**
    - **Take ownership**

    Note: Select only the actions that you need to monitor to meet your security requirements. For more information on these auditable events, see your Windows documentation.

11. If you do not want the auditing setting to propagate to subsequent files and folders of the original container, select **Apply these auditing entries to objects and/or containers within this container only** box.

12. Click **Apply**.

13. After you finish adding, removing, or editing auditing entries, click **OK**.

    The Auditing Entry for <object> box closes.

14. In the **Auditing** box, select the inheritance settings for this folder.

    You can choose one of the following:

    - Select the **Include inheritable auditing entries from this object's parent** box.
    - Select the **Replace all existing inheritable auditing entries on all descendants with inheritable auditing entries from this object** box.
    - Select both boxes.
    - Select neither box.

If you are setting SACLs on a single file, the **Replace all existing inheritable auditing entries on all descendants with inheritable auditing entries from this object** box is not present in the Auditing dialog box.

Note: Select only the minimal level that provides the auditing events that meet your security requirements.

15. Click **OK**.

    The Auditing box closes.

How to configure NTFS audit policies using the Data ONTAP CLI

You can configure audit policies on files and folders using the Data ONTAP CLI. This enables you to configure NTFS audit policies without needing to connect to the data using an SMB share on a Windows client.

You can configure NTFS audit policies by using the `vserver security file-directory` command family.

You can only configure NTFS SACLs using the CLI. Configuring NFSv4 SACLs is not supported with this Data ONTAP command family. See the man pages for more information about using these commands to configure and add NTFS SACLs to files and folders.

Configuring auditing for UNIX security style files and directories

You configure auditing for UNIX security style files and directories by adding audit ACEs to NFSv4.x ACLs. This allows you to monitor certain NFS file and directory access events for security purposes.

About this task

For NFSv4.x, both discretionary and system ACEs are stored in the same ACL. They are not stored in separate DACLs and SACLs. Therefore, you must exercise caution when adding audit ACEs to an existing ACL to avoid overwriting and losing an existing ACL. The order in which you add the audit ACEs to an existing ACL does not matter.

Steps

1. Retrieve the existing ACL for the file or directory by using the `nfs4_getfacl` or equivalent command.

   For more information about manipulating ACLs, see the man pages of your NFS client.

2. Append the desired audit ACEs.
3. Apply the updated ACL to the file or directory by using the `nfs4_setfacl` or equivalent command.

Displaying information about audit policies applied to files and directories

Displaying information about audit policies applied to files and directories enables you to verify that you have the appropriate system access control lists (SACLs) set on specified files and folders.

Displaying information about audit policies using the Windows Security tab

You can display information about audit policies that have been applied to files and directories by using the Security tab in the Windows Properties window. This is the same method used for data residing on a Windows server, which enables customers to use the same GUI interface that they are accustomed to using.

About this task

To display information about SACLs that have been applied to NTFS files and folders, complete the following steps on a Windows host.

Steps

1. From the **Tools** menu in Windows Explorer, select **Map network drive**.
2. Complete the **Map Network Drive** dialog box:
   a) Select a **Drive** letter.
   b) In the **Folder** box, type the IP address or CIFS server name of the Storage Virtual Machine (SVM) containing the share that holds both the data you would like to audit and the name of the share.

      #### Example

      If your CIFS server name is “CIFS_SERVER” and your share is named “share1”, you should enter `\\CIFS_SERVER\share1`.

      Note: You can specify the IP address of the data interface for the CIFS server instead of the CIFS server name.

   c) Click **Finish**.

   The drive you selected is mounted and ready with the Windows Explorer window displaying files and folders contained within the share.
3. Select the file or directory for which you display auditing information.
4. Right-click on the file or directory, and select **Properties**.
5. Select the **Security** tab.
6. Click **Advanced**.

7. Select the **Auditing** tab.
8. Click **Continue**.

   The Auditing box opens. The **Auditing entries** box displays a summary of users and groups that have SACLs applied to them.
9. In the **Auditing entries** box select the user or group whose SACL entries you want displayed.
10. Click **Edit**.

    The Auditing entry for <object> box opens.
11. In the **Access** box, view the current SACLs that are applied to the selected object.
12. Click **Cancel** to close the **Auditing entry for <object>** box.
13. Click **Cancel** to close the **Auditing** box.

Displaying information about NTFS audit policies on FlexVol volumes using the CLI

You can display information about NTFS audit policies on FlexVol volumes, including what the security styles and effective-security styles are, what permissions are applied, and information about system access control lists. You can use the results to validate your security configuration or to troubleshoot auditing issues.

About this task

You must supply the name of the Storage Virtual Machine (SVM) that contains the path to the files or directories whose audit information you want to display. If you want to customize the output, you can use the following optional parameters to display information only about file and directory security that matches the specified parameters:

Optional parameter	Description
`-fields` *`fieldsname`*, ...	You can use this parameter to display information on the fields you specify. You can use this parameter either alone or in combination with other optional parameters.
`-instance`	Displays detailed information about all entries.
`-volume-name` *`volume_name`*	Displays information where the specified path is relative to the specified volume. If this parameter is not specified, the SVM root volume is taken as default.
`-share-name` *`share_name`*	Displays information where the specified path is relative to the root of the specified share. If this parameter is not specified, the SVM root volume is taken as default.

Optional parameter	Description
`-lookup-names {true|false}`	Displays information where the information about owner and group is set to one of the following: • `true` displays information where the lookup name is stored as a name. • `false` displays information where the lookup name is stored as a SID.
`-expand-mask {true|false}`	Displays information where the hexadecimal bit mask entry is set to one of the following: • `true` displays information where the bit mask entries are store in expanded form. • `false` displays information where the bit mask entries are store in collapsed form.
`-security-style {unix|ntfs|mixed|unified}`	Displays information for files and directories with paths in volumes of the specified security style. This command is not supported for SVMs with Infinite Volumes; therefore, the `unified` value is not valid for this release. This is the associated security type of the volume or qtree.
`-effective-style {unix|ntfs|mixed|unified}`	Displays information for files and directories with the specified effective security style on the path. This command is not supported for SVMs with Infinite Volumes; therefore, the `unified` value is not valid for this release. This is the security scheme in effect for a given file or directory. A file or directory can have one of two security styles, either NTFS or UNIX. The effective security style is important with mixed security-style volumes and qtrees since a file or directory can have either NTFS-effective or UNIX-effective security (but not both).
`-dos-attributes` *hex_integer*	Displays information only for files and directories with the specified DOS attributes.
`-text-dos-attr` *text*	Displays information only for files and directories with the specified text DOS attributes.
`-expanded-dos-attr` *text*	Displays information only for files and directories with the specified extended DOS attributes.
`-user-id` *unix_user_ID*	Displays information only for files and directories with the specified UNIX user ID.
`-group-id` *unix_group_ID*	Displays information only for files and directories with the specified UNIX group ID.

Optional parameter	Description
`-mode-bits` *`octal_permissions`*	Displays information only for files and directories with the specified UNIX mode bits in Octal form.
`-text-mode-bits` *`text`*	Displays information only for files and directories with the specified UNIX mode bits in text form.
`-acls` *`system_acls`*	Displays information only for files and directories with the specified ACLs. You can enter the following information: • Type of ACL, which can be NTFS or NFSv4 • Control bits in the security descriptors • Owner, which applies only in the case of NTFS security descriptors. • Group, which applies only in the case of NTFS security descriptors. • Access Control Entries (ACEs) which includes both discretionary access control list (DACL) and system access control list (SACL) access control entries (ACEs) in the ACL.

Note: NTFS security-style volumes and qtrees use only NTFS system access control lists for audit policies. Mixed security-style volumes and qtrees can contain some files and directories that are of NTFS security style, which can have NTFS audit policies applied to them.

Step

1. Display audit policy settings:

   `vserver security file-directory show -vserver` ***`vserver_name`*** **`-path`** ***`path optional_parameters`***

   #### Example

   The following example displays the audit policy information about the path `/corp` in SVM vs1. This NTFS-security-style path has a NTFS-effective security style. The NTFS security descriptor contains both a SUCCESS and a SUCCESS/FAIL SACL entry:

   `vserver security file-directory show -vserver vs1 -path /corp`

```
         Vserver: vs1
              File Path: /corp
        Security Style: ntfs
       Effective Style: ntfs
        DOS Attributes: 10
 DOS Attributes in Text: ----D---
Expanded Dos Attributes: -
          Unix User Id: 0
         Unix Group Id: 0
        Unix Mode Bits: 777
 Unix Mode Bits in Text: rwxrwxrwx
                  ACLs: NTFS Security Descriptor
                        Control:0x8014
                        Owner:DOMAIN\Administrator
                        Group:BUILTIN\Administrators
                        SACL - ACEs
```

```
                ALL-DOMAIN\Administrator-0x100081-OI|CI|SA|FA
                SUCCESSFUL-DOMAIN\user1-0x100116-OI|CI|SA
              DACL - ACEs
                ALLOW-BUILTIN\Administrators-0x1f01ff-OI|CI
                ALLOW-BUILTIN\Users-0x1f01ff-OI|CI
                ALLOW-CREATOR OWNER-0x1f01ff-OI|CI
                ALLOW-NT AUTHORITY\SYSTEM-0x1f01ff-OI|CI
```

Managing auditing configurations

You can manage Storage Virtual Machine (SVM) auditing configurations by manually rotating the audit logs, enabling or disabling auditing, displaying information about auditing configurations, modifying auditing configurations, and deleting auditing configurations. You also need to understand what happens when reverting to a release where auditing is not supported.

Manually rotating the audit event logs

Before you can view the audit event logs, the logs must be converted to user-readable formats. If you want to view the event logs for a specific Storage Virtual Machine (SVM) before Data ONTAP automatically rotates the log, you can manually rotate the audit event logs on an SVM.

Step

1. Rotate the audit event logs by using the `vserver audit rotate-log` command.

Example

```
vserver audit rotate-log -vserver vs1
```

The audit event log is saved in the SVM audit event log directory with the format specified by the auditing configuration (`XML` or `EVTX`), and can be viewed by using the appropriate application.

Enabling and disabling auditing on SVMs

You can enable or disable auditing on Storage Virtual Machines (SVMs) with FlexVol volumes. You might want to temporarily stop file and directory auditing by disabling auditing. You can enable auditing at any time (if an auditing configuration exists).

Before you begin

The Storage Virtual Machine (SVM) auditing configuration must already exist before you enable auditing. Disabling auditing does not delete the auditing configuration.

Steps

1. Perform the appropriate command:

If you want auditing to be...	Enter the command...
Enabled	**vserver audit enable -vserver *vserver_name***
Disabled	**vserver audit disable -vserver *vserver_name***

2. Verify that auditing is in the desired state:

   vserver audit show -vserver *vserver_name*

Examples

The following example enables auditing for SVM vs1:

```
cluster1::> vserver audit enable -vserver vs1

cluster1::> vserver audit show -vserver vs1

 Vserver     State  Log Format Target Directory
 ----------- ------ ---------- --------------------
 vs1         true   evtx       /audit_log
```

The following example disables auditing for SVM vs1:

```
cluster1::> vserver audit disable -vserver vs1

 Vserver     State  Log Format Target Directory
 ----------- ------ ---------- --------------------
 vs1         false  evtx       /audit_log
```

Displaying information about auditing configurations

You can display information about auditing configurations for Storage Virtual Machines (SVMs) with FlexVol volumes. The information can help you determine whether the configuration is what you want in place for each SVM. The displayed information also enables you to verify whether an auditing configuration is enabled.

About this task

You can display detailed information about auditing configurations on all SVMs or you can customize what information is displayed in the output by specifying optional parameters. If you do not specify any of the optional parameters, the following is displayed:

- SVM name to which the auditing configuration applies
- The audit state, which can be `true` or `false`

  If the audit state is `true`, auditing is enabled. If the audit state is `false`, auditing is disabled.
- The audit log format
- The target directory where the auditing subsystem stores consolidated and converted audit logs

Step

1. Display information about the auditing configuration by using the `vserver audit show` command.

   For more information about using the command, see the man pages.

Examples

The following example displays the name, audit state, and target directory for all SVMs:

```
cluster1::> vserver audit show

 Vserver     State  Log Format Target Directory
 ----------- ------ ---------- --------------------
 vs1         false  evtx       /audit_log
```

The following example displays SVM names and details about the audit log for all SVMs:

```
cluster1::> vserver audit show -log-save-details

             Rotation                            Rotation
 Vserver     File Size Rotation Schedule         Limit
 ----------- --------- ------------------------- --------
 vs1         100MB     -                         0
```

The following example displays, in list form, all audit configuration information about all SVMs:

```
cluster1::> vserver audit show -instance

                           Vserver: vs1
                     Auditing state: true
                Log Destination Path: /audit_log
                          Log Format: evtx
                 Log File Size Limit: 100MB
       Log Rotation Schedule: Month: -
  Log Rotation Schedule: Day of Week: -
         Log Rotation Schedule: Day: -
        Log Rotation Schedule: Hour: -
      Log Rotation Schedule: Minute: -
                  Rotation Schedules: -
            Log Files Rotation Limit: 0
```

Commands for modifying auditing configurations

If you want to change an auditing setting for your Storage Virtual Machine (SVM), you can modify the current configuration at any time.

If you want to...	Use this command...
Modify the log destination path	`vserver audit modify` with the `-destination` parameter
Enabling automatic saves based on internal log file size	`vserver audit modify` with the `-rotate-size` parameter
Enabling automatic saves based on a time interval	`vserver audit modify` with the `-rotate-schedule-month`, `-rotate-schedule-dayofweek`, `-rotate-schedule-day`, `-rotate-schedule-hour`, and `-rotate-schedule-minute` parameters
Specifying the maximum number of saved log files	`vserver audit modify` with the `-rotate-limit` parameter

See the man page for the `vserver audit modify` command for more information.

Deleting an auditing configuration

In you no longer want to audit file and directory events on the Storage Virtual Machine (SVM) and do not want to maintain an auditing configuration on the SVM, you can delete the auditing configuration.

Steps

1. Disable the auditing configuration:

   `vserver audit disable -vserver` ***`vserver_name`***

   Example

   ```
   vserver audit disable -vserver vs1
   ```

2. Delete the auditing configuration:

   `vserver audit delete -vserver` ***`vserver_name`***

   Example

   ```
   vserver audit delete -vserver vs1
   ```

What the process is when reverting

If you plan to revert the cluster you should be aware of the process Data ONTAP follows when reverting and there are auditing-enabled Storage Virtual Machines (SVMs) in the cluster. You must take certain actions before reverting.

Reverting to a version of Data ONTAP that supports auditing, but does not support the EVTX log format

Support for the `EVTX` log format starts with Data ONTAP 8.2.1 in the 8.2 release family. If you are reverting to Data ONTAP 8.2, a version that supports auditing, but does not support the `EVTX` log format, you do not need to disable auditing on auditing-enabled SVMs before you revert. However, for each auditing configuration on the cluster (enabled or disabled), you must change the log format to the `XML` log format prior to reverting.

Reverting to a version of Data ONTAP that does not supports auditing

Support for auditing starts with Data ONTAP 8.2. If you plan to revert the cluster to a Data ONTAP release that does not support auditing and you have audit-enabled Storage Virtual Machines (SVMs), you should be aware of the process Data ONTAP follows when reverting.

- Prior to revert, you must manually disable and delete all auditing configurations on all SVMs in the cluster.

  When you disable auditing on all SVMs in the cluster, Data ONTAP consolidates and converts all auditing logs in the staging files for all SVMs. All converted audit logs are stored in the event log directory location specified in the auditing configuration for each audit-enabled SVM. The converted event logs are available post-revert.
- When you delete all auditing configurations across the cluster, Data ONTAP deletes all staging volumes.

  There is no need to manually delete staging volumes.
- During the revert, each file that has an NFSv4.x ACL is checked to determine whether the ACL contains an audit ACE.

  If it does, the complete ACL is dropped.

Troubleshooting auditing and staging volume space issues

Issues can arise when there is insufficient space on either the staging volumes or on the volume containing the audit event logs. If there is insufficient space, new audit records cannot be created, which prevents clients from accessing data, and access requests fail. You should know how to troubleshoot and resolve these volume space issues.

How to troubleshoot space issues related to the event log volumes

If volumes containing event log files run out of space, auditing cannot convert log records into log files. This results in client access failures. You need to know how to troubleshoot space issues related to event log volumes.

- Storage Virtual Machine (SVM) and cluster administrators can determine whether there is insufficient volume space by displaying information about volume and aggregate usage and configuration.
- If there is insufficient space in the volumes containing event logs, SVM and cluster administrators can resolve the space issues by either removing some of the event log files or by increasing the size of the volume.

  Note: If the aggregate that contains the event log volume is full, then the size of the aggregate must be increased before you can increase the size of the volume. Only a cluster administrator can increase the size of an aggregate.
- The destination path for the event log files can be changed to a directory on another volume by modifying the auditing configuration.

For more information about viewing information about volumes and increasing volume size, see the *Clustered Data ONTAP Logical Storage Management Guide*.

For more information about viewing information about aggregates and managing aggregates, see the *Clustered Data ONTAP Physical Storage Management Guide*.

How to troubleshoot space issues related to the staging volumes (cluster administrators only)

If any of the volumes containing staging files for your Storage Virtual Machine (SVM) runs out of space, auditing cannot write log records into staging files. This results in client access failures. To troubleshoot this issue, a cluster administrator needs to determine whether any of the staging volumes used in the SVM are full by displaying information about volume usage.

If the volume containing the consolidated event log files has sufficient space but there are still client access failures due to insufficient space, then the staging volumes might be out of space. The SVM administrator must contact the cluster administrator to determine whether the staging volumes that contain staging files for the SVM have insufficient space. The auditing subsystem generates an EMS event if auditing events cannot be generated due to insufficient space in a staging volume. The following message is displayed: `No space left on device`. Only the cluster administrator can view information about staging volumes.

If there is insufficient space in the staging volumes, the cluster administrators can resolve the space issues by increasing the size of the volume.

Note: If the aggregate that contains the staging volume is full, then the size of the aggregate must be increased before the cluster administrator can increase the size of the volume. Only a cluster administrator can increase the size of an aggregate.

For more information about viewing information about volumes and increasing volume size, see the *Clustered Data ONTAP Logical Storage Management Guide*.

For more information about viewing information about aggregates and managing aggregates, see the *Clustered Data ONTAP Physical Storage Management Guide*.

Using FPolicy for file monitoring and management on SVMs with FlexVol volumes

FPolicy is a file access notification framework that is used to monitor and manage file access events on Storage Virtual Machines (SVMs) with FlexVol volumes.

The framework generates notifications that are sent to either external FPolicy servers or to Data ONTAP. FPolicy supports event notifications for files and directories that are accessed using NFS and SMB.

Note: FPolicy is not supported on SVM with Infinite Volume.

How FPolicy works

Before you plan and create your FPolicy configuration, you should understand the basics of how FPolicy works.

What the two parts of the FPolicy solution are

There are two parts to an FPolicy solution. There is the Data ONTAP FPolicy framework that manages activities on the cluster and sends notifications to external FPolicy servers and there are external FPolicy servers that process notifications sent by Data ONTAP FPolicy.

The Data ONTAP framework creates and maintains the FPolicy configuration, monitors file events, and sends notifications to external FPolicy servers. Data ONTAP FPolicy provides the infrastructure that allows communication between external FPolicy servers and Storage Virtual Machine (SVM) nodes.

The FPolicy framework connects to external FPolicy servers and sends notifications for certain file system events to the FPolicy servers when these events occur as a result of client access. The external FPolicy servers process the notifications and send responses back to the node. What happens as a result of the notification processing depends on the application and whether the communication between the node and the external servers is asynchronous or synchronous.

What synchronous and asynchronous communications are

FPolicy sends notifications to external FPolicy servers via the FPolicy interface. The notifications are sent either in synchronous or asynchronous mode. The notification mode determines what Data ONTAP does after sending notifications to FPolicy servers.

Asynchronous notifications With asynchronous notifications, the node does not wait for a response from the FPolicy server, which enhances overall throughput of the system. This type of notification is suitable for applications where the FPolicy server does not require

that any action be taken as a result of notification evaluation. For example, asynchronous notifications are used when the Storage Virtual Machine (SVM) administrator wants to monitor and audit file access activity.

Synchronous notifications When configured to run in synchronous mode, the FPolicy server must acknowledge every notification before the client operation is allowed to continue. This type of notification is used when an action is required based on the results of notification evaluation. For example, synchronous notifications are used when the SVM administrator wants to either allow or deny requests based on criteria specified on the external FPolicy server.

Synchronous and asynchronous applications

There are many possible uses for FPolicy applications, both asynchronous and synchronous.

Asynchronous applications are ones where the external FPolicy server does not alter access to files or directories or modify data on the Storage Virtual Machine (SVM) and include the following:

- File access and audit logging
- Storage resource management

Synchronous uses cases are ones where data access is altered or data is modified by the external FPolicy server and include the following:

- Quota management
- File access blocking
- File archiving and hierarchical storage management
- Encryption and decryption services
- Compression and decompression services

These applications are in no way all-encompassing, and by using the SDK for FPolicy, implementation of other applications are possible.

Roles that cluster components play with FPolicy

The cluster, the contained Storage Virtual Machines (SVMs), and data LIFs all play a role in an FPolicy implementation.

cluster The cluster contains the FPolicy management framework and maintains and manages information about all FPolicy configurations in the cluster.

SVM An FPolicy configuration is defined at the SVM level. The scope of the configuration is the SVM, and it only operates on SVM resources. One SVM configuration cannot monitor and send notifications for file access requests that are made for data residing on another SVM.

FPolicy configurations can be defined on the admin SVM. Once configurations are defined on the admin SVM, they can be seen and used in all SVMs.

data LIFs	Connections to the FPolicy servers are made through data LIFs belonging to the SVM with the FPolicy configuration. The data LIFs used for these connections can fail over in the same manner as data LIFs used for normal client access.

How FPolicy works with external FPolicy servers

After FPolicy is configured and enabled on the Storage Virtual Machine (SVM), FPolicy runs on every node on which the SVM participates. FPolicy is responsible for establishing and maintaining connections with external FPolicy servers (FPolicy servers), for notification processing, and for managing notification messages to and from FPolicy servers.

Additionally, as part of connection management, FPolicy has the following responsibilities:

- Ensures that file notification flows through the correct LIF to the FPolicy server.
- Ensures that when multiple FPolicy servers are associated with a policy, load balancing is done when sending notifications to the FPolicy servers.
- Attempts to reestablish the connection when a connection to an FPolicy server is broken.
- Sends the notifications to FPolicy servers over an authenticated session.

How control channels are used for FPolicy communication

FPolicy initiates a control channel connection to an external FPolicy server from the data LIFs of each node participating on a Storage Virtual Machine (SVM). FPolicy uses control channels for transmitting file notifications; therefore, an FPolicy server might see multiple control channel connections based on SVM topology.

How privileged data access channels are used for synchronous communication

With synchronous use cases, the FPolicy server accesses data residing on the Storage Virtual Machine (SVM) through a privileged data access path. Access through the privileged path exposes the complete file system to the FPolicy server. It can access data files to collect information, to scan files, read files, or write into files.

Because the external FPolicy server can access the entire file system from the root of the SVM through the privileged data channel, the privileged data channel connection must be secure.

How FPolicy connection credentials are used with privileged data access channels

The FPolicy server makes privileged data access connections to cluster nodes by using a specific Windows user credential that is saved with the FPolicy configuration. SMB is the only supported protocol for making a privileged data access channel connection.

If the FPolicy server requires privileged data access, the following conditions must be met:

- A CIFS license must be enabled on the cluster.
- The FPolicy server must run under the credentials configured in the FPolicy configuration.

When making a data channel connection, FPolicy uses the credential for the specified Windows user name. Data access is made over the admin share ONTAP_ADMIN$.

What granting super user credentials for privileged data access means

Data ONTAP uses the combination of the IP address and the user credential configured in the FPolicy configuration to grant super user credentials to the FPolicy server.

Super user status grants these privileges when the FPolicy server accesses data:

- Avoid permission checks
  The user avoids checks on files and directory access.
- Special locking privileges
  Data ONTAP allows read, write, or modify access to any file regardless of existing locks. If the FPolicy server takes byte range locks on the file, it results in immediate removal of existing locks on the file.
- By-pass any FPolicy checks
  Access does not generate any FPolicy notifications.

How FPolicy manages policy processing

There might be multiple FPolicy policies assigned to your Storage Virtual Machine (SVM); each with a different priority. To create an appropriate FPolicy configuration on the SVM, it is important to understand how FPolicy manages policy processing.

Each file access request is initially evaluated to determine which policies are monitoring this event. If it is a monitored event, information about the monitored event along with interested policies is passed to FPolicy where it is evaluated. Each policy is evaluated in order of the assigned priority.

You should consider the following recommendations when configuring policies:

- When you want a policy to always be evaluated before other policies, configure that policy with a higher priority.
- If the success of requested file access operation on a monitored event is a prerequisite for a file request that is evaluated against another policy, give the policy that controls the success or failure of the first file operation a higher priority.
  For example, if one policy manages FPolicy file archiving and restore functionality and a second policy manages file access operations on the online file, the policy that manages file restoration must have a higher priority so that the file is restored before the operation managed by the second policy can be allowed.
- If you want all policies that might apply to a file access operation to be evaluated, give synchronous policies a lower priority.

You can reorder policy priorities for existing policies by modifying the policy sequence number. However, to have FPolicy evaluate policies based on the modified priority order, you must disable and reenable the policy with the modified sequence number.

What the node-to-external FPolicy server communication process is

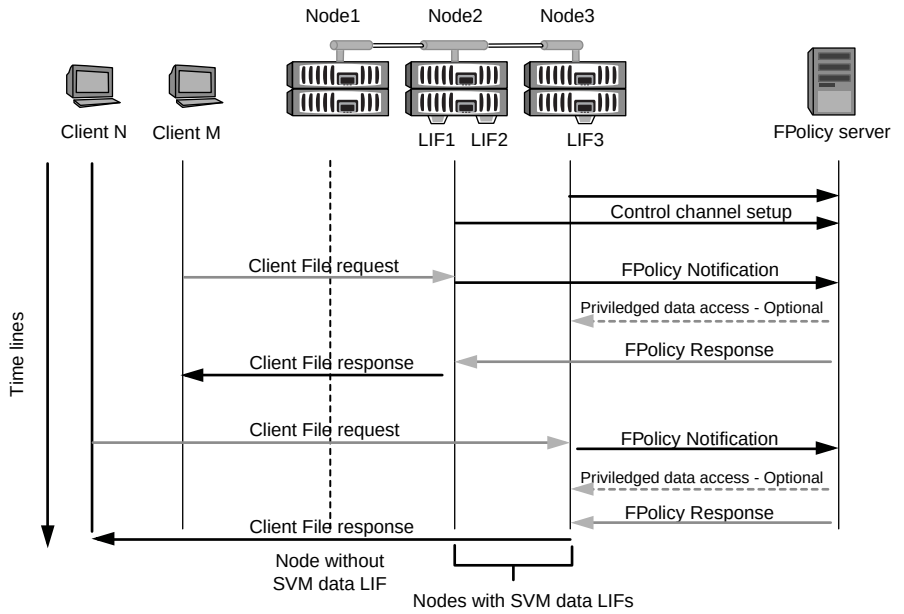
To properly plan your FPolicy configuration, you should understand what the node-to-external Fpolicy server communication process is.

Every node that participates on each Storage Virtual Machine (SVM) initiates a connection to an external FPolicy server (FPolicy server) using TCP/IP. Connections to the FPolicy servers are set up using node data LIFs; therefore, a participating node can set up a connection only if the node has an operational data LIF for the SVM.

Each FPolicy process on participating nodes attempts to establish a connection with the FPolicy server when the policy is enabled. It uses the IP address and port of the FPolicy external engine specified in the policy configuration.

The connection establishes a control channel from each of the nodes participating on each SVM to the FPolicy server through the data LIF. In addition, if IPv4 and IPv6 data LIF addresses are present on the same participating node, FPolicy attempts to establish connections for both IPv4 and IPv6. Therefore, in a scenario where the SVM extends over multiple nodes or if both IPv4 and IPv6 addresses are present, the FPolicy server must be ready for multiple control channel setup requests from the cluster after the FPolicy policy is enabled on the SVM.

For example, if a cluster has three nodes—Node1, Node2, and Node3—and SVM data LIFs are spread across only Node2 and Node3, control channels are initiated only from Node2 and Node3, irrespective of the distribution of data volumes. Say that Node2 has two data LIFs—LIF1 and LIF2—that belong to the SVM and that the initial connection is from LIF1. If LIF1 fails, FPolicy attempts to establish a control channel from LIF2.



How FPolicy manages external communication during LIF migration or failover

Data LIFs can be migrated to data ports in the same node or to data ports on a remote node.

When a data LIF fails over or is migrated, a new control channel connection is made to the FPolicy server. FPolicy can then retry SMB and NFS client requests that timed out, with the result that new notifications are sent to the external FPolicy servers. The node rejects FPolicy server responses to original, timed-out SMB and NFS requests.

How FPolicy manages external communication during node failover

If the cluster node that hosts the data ports used for FPolicy communication fails, Data ONTAP breaks the connection between the FPolicy server and the node.

The impact of cluster failover to the FPolicy server can be mitigated by configuring the LIF manager to migrate the data port used in FPolicy communication to another active node. After the migration is complete, a new connection is established using the new data port.

If the LIF manager is not configured to migrate the data port, the FPolicy server must wait for the failed node to come up. After the node is up, a new connection is initiated from that node with a new Session ID.

Note: The FPolicy server detects broken connections with the keep-alive protocol message. The timeout for purging the session ID is determined when configuring FPolicy. The default keep-alive timeout is two minutes.

How FPolicy services work across SVM namespaces

Data ONTAP provides a unified Storage Virtual Machine (SVM) namespace. Volumes across the cluster are joined together by junctions to provide a single, logical file system. The FPolicy server is aware of the namespace topology and provides FPolicy services across the namespace.

The namespace is specific to and contained within the SVM; therefore, you can see the namespace only from the SVM context. Namespaces have the following characteristics:

- A single namespace exists in each SVM, with the root of the namespace being the root volume, represented in the namespace as slash (/).
- All other volumes have junction points below the root (/).
- Volume junctions are transparent to clients.
- A single NFS export can provide access to the complete namespace; otherwise, export policies can export specific volumes.
- SMB shares can be created on the volume or on qtrees within the volume or on any directory within the namespace.
- The namespace architecture is flexible.

  Examples of typical namespace architectures are as follows:

  - A namespace with a single branch off of the root
  - A namespace with multiple branches off of the root

- A namespace with multiple unbranched volumes off of the root

FPolicy configuration types

There are two basic FPolicy configuration types. One configuration uses external FPolicy servers to process and act upon notifications. The other configuration does not use external FPolicy servers; instead, it uses the Data ONTAP internal, native FPolicy server for simple file blocking based on extensions.

External FPolicy server configuration	The notification is sent to the FPolicy server, which screens the request and applies rules to determine whether the node should allow the requested file operation. For synchronous policies, the FPolicy server then sends a response to the node to either allow or block the requested file operation.
Native FPolicy server configuration	The notification is screened internally. The request is allowed or denied based on file extensions settings configured in the FPolicy scope.

When to create a native FPolicy configuration

Native FPolicy configurations use the Data ONTAP internal FPolicy engine to monitor and block file operations based on the file's extension. This solution does not require external FPolicy servers (FPolicy servers). Using a native file blocking configuration is appropriate when this simple solution is all that is needed.

Native file blocking enables you to monitor any file operations that match configured operation and filtering events and then deny access to files with particular extensions. This is the default configuration.

This configuration provides a means to block file access based only on the file's extension. For example, to block files that contain `mp3` extensions, you configure a policy to provide notifications for certain operations with target file extensions of `mp3`. The policy is configured to deny `mp3` file requests for operations that generate notifications.

The following applies to native FPolicy configurations:

- The same set of filters and protocols that are supported by FPolicy server-based file screening are also supported for native file blocking.
- Native file blocking and FPolicy server-based file screening applications can be configured at the same time.

  To do so, you can configure two separate FPolicy policies for the Storage Virtual Machine (SVM), with one configured for native file blocking and one configured for FPolicy server-based file screening.
- The native file blocking feature only screens files based on the extensions and not on the content of the file.
- In the case of symbolic links, native file blocking uses the file extension of the root file.

When to create a configuration that uses external FPolicy servers

FPolicy configurations that use external FPolicy servers to process and manage notifications provide robust solutions for use cases where more than simple file blocking based on file extension is needed.

You should create a configuration that uses external FPolicy servers when you want to do such things as monitor and record file access events, provide quota services, perform file blocking based on criteria other than simple file extensions, provide data migration services using hierarchical storage management applications, or provide a fine-grained set of policies that monitor only a subset of data in the Storage Virtual Machine (SVM).

Requirements, considerations, and best practices for configuring FPolicy

Before you create and configure FPolicy configurations on your Storage Virtual Machines (SVMs) with FlexVol volumes, you need to be aware of certain requirements, considerations, and best practices for configuring FPolicy.

Ways to configure FPolicy

FPolicy features are configured either through the command line interface (CLI) or through APIs. This guide uses the CLI to create, manage, and monitor an FPolicy configuration on the cluster.

Requirements for setting up FPolicy

Before you configure and enable FPolicy on your Storage Virtual Machine (SVM), you need to be aware of certain requirements.

- All nodes in the cluster must be running a version of Data ONTAP that supports FPolicy.
- If you are not using the Data ONTAP native FPolicy engine, you must have external FPolicy servers (FPolicy servers) installed.
- The FPolicy servers must be installed on a server accessible from the data LIFs of the SVM where FPolicy policies are enabled.
- The IP address of the FPolicy server must be configured as a primary or secondary server in the FPolicy policy external engine configuration.
- If the FPolicy servers access data over a privileged data channel, the following additional requirements must be met:
  - CIFS must be licensed on the cluster.
    Privileged data access is accomplished using SMB connections.
  - A user credential must be configured for accessing files over the privileged data channel.
  - The FPolicy server must run under the credentials configured in the FPolicy configuration.

Best practices and recommendations when setting up FPolicy

When setting up FPolicy on Storage Virtual Machines (SVMs), you need to be familiar with configuration best practices and recommendations to ensure that your FPolicy configuration provides robust monitoring performance and results that meet your requirements.

- External FPolicy servers (FPolicy servers) should be placed in close proximity to the cluster with high-bandwidth connectivity to provide minimal latency and high-bandwidth connectivity.
- The FPolicy external engine should be configured with more than one FPolicy server to provide resiliency and high availability of FPolicy server notification processing, especially if policies are configured for synchronous screening.
- It is recommended to disable the FPolicy policy before making any configuration changes.

  For example, if you want to add or modify an IP address in the FPolicy external engine configured for the enabled policy, you should first disable the policy.
- If you configure FPolicy to monitor FlexCache volumes, it is recommended that you do not configure FPolicy to monitor `read` and `get attr` file operations on the FlexCache volumes.

  This is because Data ONTAP needs to retrieve inode-to-path (I2P) data with these operations, and this data cannot be retrieved from the FlexCache volume. Instead, the I2P request is forwarded to the origin volume, with the result that the performance benefits from FlexCache are not realized when FPolicy is used to monitor read and get attr operations on FlexCache volumes.
- The cluster node-to-FPolicy server ratio should be optimized to ensure that FPolicy servers are not overloaded, which can introduce latencies when the SVM responds to client requests.

  The optimal ratio depends on the application for which the FPolicy server is being used.

Important revert considerations

You must understand and act on some important revert considerations before reverting to a Data ONTAP release that does not support FPolicy.

Before reverting to a version of Data ONTAP that does not support FPolicy, the following conditions must be met:

- Every file on which FPolicy servers set the offline bit must be either deleted or replaced with the original files before disabling FPolicy and reverting to a version of Data ONTAP that does not support FPolicy.

  Note: If you do not replace the files with the offline bit set with the original files prior to reverting, clients access the stub files instead of the files to which the stub refers.
- FPolicy functionality must be disabled on the cluster by disabling every FPolicy policy on the cluster.

What the steps for setting up an FPolicy configuration are

Before FPolicy can monitor file access, an FPolicy configuration must be created and enabled on the Storage Virtual Machine (SVM) for which FPolicy services are required.

The steps for setting up and enabling an FPolicy configuration on the SVM are as follows:

1. Create an FPolicy external engine.
   The FPolicy external engine identifies the external FPolicy servers (FPolicy servers) that are associated with a specific FPolicy configuration. If the internal “native” FPolicy engine is used to create a native file-blocking configuration, you do not need to create an FPolicy external engine.
2. Create an FPolicy event.
   An FPolicy event describes what the FPolicy policy should monitor. Events consist of the protocols and file operations to monitor, and can contain a list of filters. Events use filters to narrow the list of monitored events for which the FPolicy external engine must send notifications. Events also specify whether the policy monitors volume operations.
3. Create an FPolicy policy.
   The FPolicy policy is responsible for associating, with the appropriate scope, the set of events that need to be monitored and for which of the monitored events notifications must be sent to the designated FPolicy server (or to the native engine if no FPolicy servers are configured). The policy also defines whether the FPolicy server is allowed privileged access to the data for which it receives notifications. An FPolicy server needs privileged access if the server needs to access the data. Typical use cases where privileged access is needed include file blocking, quota management, and hierarchical storage management. The policy is where you specify whether the configuration for this policy uses an FPolicy server or the internal “native” FPolicy server.
   A policy specifies whether screening is mandatory. If screening is mandatory and all FPolicy servers are down or no response is received from the FPolicy servers within a defined timeout period, then file access is denied.
   A policy's boundaries are the SVM. A policy cannot apply to more than one SVM. However, a specific SVM can have multiple FPolicy policies, each with the same or different combination of scope, event, and external server configurations.
4. Configure the policy scope.
   The FPolicy scope determines which volumes, shares, or export-policies the policy acts on or excludes from monitoring. A scope also determines which file extensions should be included or excluded from FPolicy monitoring.

   Note: Exclude lists take precedence over include lists.

5. Enable the FPolicy policy.
   When the policy is enabled, the control channels and, optionally, the privileged data channels are connected. The FPolicy process on the nodes on which the SVM participates begin monitoring file and folder access and, for events that match configured criteria, sends notifications to the FPolicy servers (or to the native engine if no FPolicy servers are configured).

Note: If the policy uses native file blocking, an external engine is not configured or associated with the policy.

Planning the FPolicy configuration

Before you create an FPolicy configuration, you must understand what is involved in each step of the configuration. You need to decide what settings you need to use when performing the configuration and record them in the planning worksheets.

You need to plan for the following configuration tasks:

- Creating the FPolicy external engine
- Creating the FPolicy policy event
- Creating the FPolicy policy
- Creating the FPolicy policy scope

FPolicy is supported on Storage Virtual Machines (SVMs) with FlexVol volumes. FPolicy is not supported on SVMs with Infinite Volume.

Planning the FPolicy external engine configuration

Before you configure the FPolicy external engine (external engine), you must understand what it means to create an external engine and which configuration parameters are available. This information helps you to determine which values to set for each parameter.

What it means to create an external engine

Creating the external engine means defining the information that FPolicy needs to make and manage connections to the external FPolicy servers (FPolicy servers). The external engine configuration defines the following configuration information:

- Storage Virtual Machine (SVM) name
- Engine name
- The IP addresses of the primary and secondary FPolicy servers and the TCP port number to use when making the connection to the FPolicy servers
- Whether the engine type is asynchronous or synchronous
- How to authenticate the connection between the node and the FPolicy server

  If you choose to configure mutual SSL authentication, then you must also configure parameters that provide SSL certificate information.
- How to manage the connection (advanced privilege settings)

  This includes parameters that define such things as timeout values, retry values, keep-alive values, and maximum request values.

What the basic external engine parameters are

You can use the following table of basic FPolicy configuration parameters to help you plan your configuration:

Type of information	Option
SVM Specifies the SVM name that you want to associate with this external engine. Each FPolicy configuration is defined within a single SVM. The external engine, policy event, policy scope, and policy that combine together to create an FPolicy policy configuration must all be associated with the same SVM.	`-vserver` *`vserver_name`*
Engine name Specifies the name to assign to the external engine configuration. You must specify the engine name later when you create the FPolicy policy. This associates the external engine with the policy.	`-engine-name` *`engine_name`*
Primary FPolicy servers Specifies the primary FPolicy servers to which the node sends notifications for a given FPolicy policy. The value is specified as a comma-delimited list of IP addresses. If more than one primary server IP address is specified, every node on which the SVM participates creates a control connection to every specified primary FPolicy server at the time the policy is enabled. If you configure multiple primary FPolicy servers, notifications are sent to the FPolicy servers in a round-robin fashion.	`-primary-servers` *`IP_address`*,...
Port number Specifies the port number of the FPolicy service.	`-port` *`integer`*
Secondary FPolicy servers Specifies the secondary FPolicy servers to which to send file access events for a given FPolicy policy. The value is specified as a comma-delimited list of IP addresses. Secondary servers are used only when none of the primary servers are reachable. Connections to secondary servers are established when the policy is enabled, but notifications are sent to secondary servers only if none of the primary servers are reachable. If you configure multiple secondary servers, notifications are sent to the FPolicy servers in a round-robin fashion.	`-secondary-servers` *`IP_address`*,...

Type of information	Option
External engine type Specifies whether the external engine operates in synchronous or asynchronous mode. By default, FPolicy operates in synchronous mode. When set to `synchronous`, file request processing sends a notification to the FPolicy server, but then does not continue until after receiving a response from the FPolicy server. At that point, request flow either continues or processing results in denial, depending on whether the response from the FPolicy server permits the requested action. When set to `asynchronous`, file request processing sends a notification to the FPolicy server, and then continues.	`-extern-engine-type` *`external_engine_type`* The value for this parameter can be one of the following: • `synchronous` • `asynchronous`
SSL option for communication with FPolicy server Specifies the SSL option for communication with the FPolicy server. This is a required parameter. You can choose one of the options based on the following information: • When set to `no-auth`, no authentication takes place. The communication link is established over TCP. • When set to `server-auth`, the SVM authenticates the FPolicy server. If you choose this value, before creating the external engine, you must install the public certificate of the certificate authority (CA) that signed the FPolicy server certificate. • When set to `mutual-auth`, mutual authentication takes place between the SVM and the FPolicy server; the SVM authenticates the FPolicy server, and the FPolicy server authenticates the SVM. If you choose this value, before creating the external engine, the administrator must install the public certificate of the CA that signed the FPolicy server certificate along with the public certificate and key file for authentication of the SVM. The public certificate of CA that is used to sign the FPolicy server certificate is installed by using the `security certificate install` command with the `-type` parameter set to `client_ca`. The private key and public certificate required for authentication of the SVM is installed by using the `security certificate install` command with the `-type` parameter set to `server`. If you choose to configure mutual SSL authentication, then you must also configure the `-certificate-common-name`, `-certificate-serial`, and `-certifcate-ca` parameters.	`-ssl-option {no-auth|server-auth|mutual-auth}`

Type of information	Option
Certificate FQDN or custom common name Specifies the certificate name used if SSL authentication between the SVM and the FPolicy server is configured. You can specify the certificate name as an FQDN or as a custom common name. If you specify `mutual-auth` for the `-ssl-option` parameter, you must specify a value for the `-certificate-common-name` parameter.	`-certificate-common-name` *text*
Certificate serial number Specifies the serial number of the certificate used for authentication if SSL authentication between the SVM and the FPolicy server is configured. If you specify `mutual-auth` for the `-ssl-option` parameter, you must specify a value for the `-certificate-serial` parameter.	`-certificate-serial` *text*
Certificate authority Specifies the CA name of the certificate used for authentication if SSL authentication between the SVM and the FPolicy server is configured. If you specify `mutual-auth` for the `-ssl-option` parameter, you must specify a value for the `-certifcate-ca` parameter.	`-certifcate-ca` *text*

What the advanced external engine options are

You can use the following table of advanced FPolicy configuration parameters as you plan whether to customize your configuration with advanced parameters. You use these parameters to modify communication behavior between the cluster nodes and the FPolicy servers:

Type of information	Option
Timeout for canceling a request Specifies the time interval in hours (`h`), minutes (`m`), or seconds (`s`) that the node waits for a response from the FPolicy server. If the timeout interval passes, the node sends a cancel request to the FPolicy server. The node then sends the notification to an alternate FPolicy server. This timeout helps in handling an FPolicy server that is not responding, which can improve SMB/NFS client response. Also, canceling requests after a timeout period can help in releasing system resources because the notification request is moved from a down/bad FPolicy server to an alternate FPolicy server. The range for this value is `0` through `100`. If the value is set to `0`, the option is disabled and cancel request messages are not sent to the FPolicy server. The default is `20s`.	`-reqs-cancel-timeout` *integer*[h|m|s]

Type of information	Option
Timeout for aborting a request Specifies the timeout in hours (`h`), minutes (`m`), or seconds (`s`) for aborting a request. The range for this value is `0` through `200`.	`-reqs-abort-timeout` *integer*[h|m|s]
Interval for sending status requests Specifies the interval in hours (`h`), minutes (`m`), or seconds (`s`) after which a status request is sent to the FPolicy server. The range for this value is `0` through `50`. If the value is set to `0`, the option is disabled and status request messages are not sent to the FPolicy server. The default is `10s`.	`-status-req-interval` *integer*[h|m|s]
Maximum outstanding requests on the FPolicy server Specifies the maximum number of outstanding requests that can be queued on the FPolicy server. The range for this value is `1` through `10000`. The default is `50`.	`-max-server-reqs` *integer*
Timeout for disconnecting a nonresponsive FPolicy server Specifies the time interval in hours (`h`), minutes (`m`), or seconds (`s`) after which the connection to the FPolicy server is terminated. The connection is terminated after the timeout period only if the FPolicy server's queue contains the maximum allowed requests and no response is received within the this timeout period. The maximum allowed number of requests is either `50` (the default) or the number specified by the `max-server-reqs-` parameter. The range for this value is `1` through `100`. The default is `60s`.	`-server-progress-timeout` *integer*[h|m|s]
Interval for sending keep-alive messages to the FPolicy server Specifies the time interval in hours (`h`), minutes (`m`), or seconds (`s`) at which keep-alive messages are sent to the FPolicy server. Keep-alive messages detect half-open connections. The range for this value is `10` through `600`. If the value is set to `0`, the option is disabled and keep-alive messages are prevented from being sent to the FPolicy servers. The default is `120s`.	`-keep-alive-interval-` *integer*[h|m|s]
Maximum reconnect attempts Specifies the maximum number of times the SVM attempts to reconnect to the FPolicy server after the connection has been broken. The range for this value is `0` through `20`. The default is `5`.	`-max-connection-retries` *integer*

Completing the FPolicy external engine configuration worksheet

You can use this worksheet to record the values that you need during the FPolicy external engine configuration process. If a parameter value is required, you need to determine what value to use for those parameters before you configure the external engine.

Information for a basic external engine configuration

You should record whether you want to include each parameter setting in the external engine configuration and then record the value for the parameters that you want to include.

Type of information	Required	Include	Your values
Storage Virtual Machine (SVM) name	Yes	Yes	
Engine name	Yes	Yes	
Primary FPolicy servers	Yes	Yes	
Port number	Yes	Yes	
Secondary FPolicy servers	No		
External engine type	No		
SSL option for communication with external FPolicy server	Yes	Yes	
Certificate FQDN or custom common name	No		
Certificate serial number	No		
Certificate authority	No		

Information for advanced external engine parameters

To configure an external engine with advanced parameters, you must enter the configuration command while in advanced privilege mode.

Type of information	Required	Include	Your values
Timeout for canceling a request	No		
Timeout for aborting a request	No		
Interval for sending status requests	No		

Type of information	Required	Include	Your values
Maximum outstanding requests on the FPolicy server	No		
Timeout for disconnecting a nonresponsive FPolicy server	No		
Interval for sending keep-alive messages to the FPolicy server	No		
Maximum reconnect attempts	No		

Planning the FPolicy event configuration

Before you configure FPolicy events, you must understand what it means to create an FPolicy event. You must determine which protocols you want the event to monitor, which events to monitor, and which event filters to use. This information helps you plan the values that you want to set.

What it means to create an FPolicy event

Creating the FPolicy event means defining information that the FPolicy process needs to determine what file access operations to monitor and for which of the monitored events notifications should be sent to the external FPolicy server. The FPolicy event configuration defines the following configuration information:

- Storage Virtual Machine (SVM) name
- Event name
- Which protocols to monitor
  FPolicy can monitor SMB, NFSv3, and NFSv4 file access operations.
- Which file operations to monitor
  Not all file operations are valid for each protocol.
- Which file filters to configure
  Only certain combinations of file operations and filters are valid. Each protocol has its own set of supported combinations.
- Whether to monitor volume operations

  Note: There is a dependency with three of the parameters (`-protocol`, `-file-operations`, `-filters`). The following are the valid combinations for the three parameters:

  - You can specify the `-protocol` and `-file-operations` parameters.
  - You can specify all three of the parameters.
  - You can specify none of the parameters.

What the FPolicy event configuration contains

You can use the following list of available FPolicy event configuration parameters to help you plan your configuration:

Type of information	Option
SVM Specifies the SVM name that you want to associate with this FPolicy event. Each FPolicy configuration is defined within a single SVM. The external engine, policy event, policy scope, and policy that combine together to create an FPolicy policy configuration must all be associated with the same SVM.	`-vserver` *vserver_name*
Event name Specifies the name to the FPolicy event configuration. When you create the FPolicy policy you associate the FPolicy event with the policy using the event name.	`-event-name` *event_name*
Protocol Specifies which protocol to configure for the FPolicy event. The list for `-protocol` can include one of the following values: • `cifs` • `nfsv3` • `nfsv4` **Note:** If you specify `-protocol`, then you must specify a valid value in the `-file-operations` parameter. As the protocol version changes, the valid values might change.	`-protocol` *protocol*

Type of information	Option
File operations Specifies the list of file operations for the FPolicy event. The event checks the operations specified in this list from all client requests using the protocol specified in the `-protocol` parameter. You can list one or more file operations by using a comma-delimited list. The list for `-file-operations` can include one or more of the following values: • `close` for file close operations • `create` for file create operations • `create-dir` for directory create operations • `delete` for file delete operations • `delete_dir` for directory delete operations • `getattr` for get attribute operations • `link` for link operations • `lookup` for lookup operations • `open` for file open operations • `read` for file read operations • `write` for file write operations • `rename` for file rename operations • `rename_dir` for directory rename operations • `setattr` for set attribute operations • `symlink` for symbolic link operations **Note:** If you specify `-file-operations`, then you must specify a valid protocol in the `-protocol` parameter.	`-file-operations` *file_operations*,...

Type of information	Option
Filters Specifies the list of filters for a given file operation for the specified protocol. The values in the `-filters` parameter are used to filter client requests. The list can include one or more of the following: • `monitor-ads` to filter the client request for alternate data stream • `close-with-modification` to filter the client request for close with modification • `close-without-modification` to filter the client request for close without modification • `first-read` to filter the client request for first read • `first-write` to filter the client request for first write • `offline-bit` to filter the client request for offline bit set Setting this filter results in the FPolicy server receiving notification only when offline files are accessed. • `open-with-delete-intent` to filter the client request for open with delete intent Setting this filter results in the FPolicy server receiving notification only when an attempt is made to open a file with the intent to delete it. This is used by file systems when the `FILE_DELETE_ON_CLOSE` flag is specified. • `open-with-write-intent` to filter client request for open with write intent Setting this filter results in the FPolicy server receiving notification only when an attempt is made to open a file with the intent to write something in it. • `write-with-size-change` to filter the client request for write with size change **Note:** If you specify the `-filters` parameter, then you must also specify valid values for the `-file-operations` and `-protocol` parameters.	`-filters` *`filter`*, ...
Is volume operation required Specifies whether volume operation monitoring is required. The default is `false`.	`-volume-operation {true|false}`

List of supported file operation and filter combinations that FPolicy can monitor for SMB

When you configure your FPolicy event, you need to be aware that only certain combinations of file operations and filters are supported for monitoring SMB file access operations.

The list of supported file operation and filter combinations for FPolicy monitoring of SMB file access events is provided in the following table:

Supported file operations	Supported filters
close	monitor-ads, offline-bit, close-with-modification, close-without-modification
create	monitor-ads, offline-bit
create_dir	Currently no filter is supported for this file operation.
delete	monitor-ads, offline-bit
delete_dir	Currently no filter is supported for this file operation.
getattr	offline-bit
open	monitor-ads, offline-bit, open-with-delete-intent, open-with-write-intent
read	monitor-ads, offline-bit, first-read
write	monitor-ads, offline-bit, first-write, write-with-size-change
rename	monitor-ads, offline-bit
rename_dir	Currently no filter is supported for this file operation.
setattr	monitor-ads, offline-bit

List of supported file operation and filter combinations that FPolicy can monitor for NFSv3

When you configure your FPolicy event, you need to be aware that only certain combinations of file operations and filters are supported for monitoring NFSv3 file access operations.

The list of supported file operation and filter combinations for FPolicy monitoring of NFSv3 file access events is provided in the following table:

Supported file operations	Supported filters
create	offline-bit
create_dir	Currently no filter is supported for this file operation.
delete	offline-bit
delete_dir	Currently no filter is supported for this file operation.

Supported file operations	Supported filters
link	offline-bit
lookup	offline-bit
read	offline-bit
write	offline-bit, write-with-size-change
rename	offline-bit
rename_dir	Currently no filter is supported for this file operation.
setattr	offline-bit
symlink	offline-bit

List of supported file operation and filter combinations that FPolicy can monitor for NFSv4

When you configure your FPolicy event, you need to be aware that only certain combinations of file operations and filters are supported for monitoring NFSv4 file access operations.

The list of supported file operation and filter combinations for FPolicy monitoring of NFSv4 file access events is provided in the following table:

Supported file operations	Supported filters
close	offline-bit
create	offline-bit
create_dir	Currently no filter is supported for this file operation.
delete	offline-bit
delete_dir	Currently no filter is supported for this file operation.
getattr	offline-bit
link	offline-bit
lookup	offline-bit
open	offline-bit
read	offline-bit
write	offline-bit, write-with-size-change
rename	offline-bit
rename_dir	Currently no filter is supported for this file operation.
setattr	offline-bit

Supported file operations	Supported filters
symlink	offline-bit

Completing the FPolicy event configuration worksheet

You can use this worksheet to record the values that you need during the FPolicy event configuration process. If a parameter value is required, you need to determine what value to use for those parameters before you configure the FPolicy event.

You should record whether you want to include each parameter setting in the FPolicy event configuration and then record the value for the parameters that you want to include.

Type of information	Required	Include	Your values
Storage Virtual Machine (SVM) name	Yes	Yes	
Event name	Yes	Yes	
Protocol	No		
File operations	No		
Filters	No		
Is volume operation required	No		

Planning the FPolicy policy configuration

Before you configure the FPolicy policy, you must understand what it means to create an FPolicy policy. You must understand what configuration options are available. You also need to understand why you might want to attach more than one event to an FPolicy policy. This information helps you as you determine what values that you want to set.

What it means to create an FPolicy policy

Creating the FPolicy policy means associating a specific Storage Virtual Machine (SVM), an FPolicy event, and an FPolicy external engine (external engine) to an FPolicy policy. You also specify the following:

- Whether mandatory screening is required for this policy.
- Whether to use the Data ONTAP native external engine for simple file blocking or whether to specify an external engine that is configured to use external FPolicy servers (FPolicy servers) for more sophisticated file blocking and file management.
- Whether you want to associate more than one FPolicy event to the policy.
  An event is specific to a protocol. You can use a single FPolicy policy to monitor file access events for more than one protocol by creating an event for each protocol that you want the policy to monitor, and then associating the events to the policy.

- Whether you want the FPolicy server to have privileged access to the monitored files and folders by using a privileged data connection.

  If you want to configure the policy to allow privileged access, you must also specify the user name for the account that you want the FPolicy server to use for privileged access.

What the FPolicy policy configuration contains

You can use the following list of available FPolicy policy configuration parameters to help you plan your configuration:

Type of information	Option
SVM Specifies the SVM name on which you want to create an FPolicy policy. Each FPolicy configuration is defined within a single SVM. The external engine, FPolicy event, FPolicy scope, and FPolicy policy that combine together to create an FPolicy policy configuration must all be associated with the same SVM.	`-vserver` *`vserver_name`*
Policy name Specifies the name of the FPolicy policy. The name can be up to 256 characters long and is a string that can only contain any combination of ASCII-range alphanumeric characters (`a` through `z`, `A` through `Z`, and `0` through `9`), "_", and ".".	`-policy-name` *`policy_name`*
Event names Specifies a comma-delimited list of events to associate with the FPolicy policy. The events must already exist.	`-events` *`event_name`*, ...
External engine name Specifies the name of the external engine to associate with the FPolicy policy. The external engine must already exist. An external engine contains information required by the node to send notifications to an FPolicy server. The default value for this parameter is `native`. This means that, if you do not specify a value for the external engine, the default native external engine is used. The native external engine is internal to Data ONTAP and is used if you want to configure native file blocking and you do not want to use FPolicy servers. If you want to use the native external engine, you can either not specify a value for this parameter or you can specify `native` as the value.	`-engine` *`engine_name`*

Type of information	Option
Is mandatory screening required Specifies whether mandatory file access screening is required. This parameter specifies what action to take on a file access event in a case when all primary and secondary servers are down or no response is received from the FPolicy servers within a given timeout period. When set to `true`, file access events are denied. When set to `false`, file access events are allowed. The default is `true`.	`-is-mandatory {true|false}`
Allow privileged access Specifies whether the FPolicy servers can have privileged access to monitored data. With this option set to `yes`, FPolicy servers can access files from the root of the SVM containing the monitored data using the privileged data channel. The default is `no`.	`-allow-privileged-access {yes|no|}`
Privileged user name Specifies the user name of the account the FPolicy servers use for privileged data access. The value for this parameter should use the “domain\user name” format. If `-allow-privileged-access` is set to `no`, any value set for this parameter is ignored.	`-privileged-user-name` *user_name*

Completing the FPolicy policy worksheet

You can use this worksheet to record the values that you need during the FPolicy policy configuration process. If a parameter value is required, you need to determine what value to use for those parameters before you configure the FPolicy policy.

You should record whether you want to include each parameter setting in the FPolicy policy configuration and then record the value for the parameters that you want to include.

Type of information	Required	Include	Your values
Storage Virtual Machine (SVM) name	Yes	Yes	
Policy name	Yes	Yes	
Event names	Yes	Yes	
External engine name	Yes	Yes	
Is mandatory screening required	No		
Allow privileged access	No		

Type of information	Required	Include	Your values
Privileged user name	No		

Planning the FPolicy scope configuration

Before you configure the FPolicy scope, you must understand what it means to create a scope. You must understand what the scope configuration contains. You also need to understand what the scope rules of precedence are. This information can help you plan the values that you want to set.

What it means to create an FPolicy scope

Creating the FPolicy scope means defining the boundaries on which the FPolicy policy applies. The Storage Virtual Machine (SVM) is the basic boundary. When you create a scope for an FPolicy policy, you must define the FPolicy policy to which it will apply, and you must designate to which SVM you want to apply the scope.

There are a number of parameters that further restrict the scope within the specified SVM. You can restrict the scope by specifying what to include in the scope or by specifying what to exclude from the scope. After you apply a scope to an enabled policy, policy event checks get applied to the scope defined by this command.

Notifications are generated for file access events where matches are found in the “include” options. Notifications are not generated for file access events where matches are found in the “exclude” options.

The FPolicy scope configuration defines the following configuration information:

- SVM name
- Policy name
- The shares to include or exclude from what gets monitored
- The export policies to include or exclude from what gets monitored
- The volumes to include or exclude from what gets monitored
- The file extensions to include or exclude from what gets monitored
- Whether to do file extension checks on directory objects

Note: There are special considerations for the scope for a cluster FPolicy policy. The cluster FPolicy policy is a policy that the cluster administrator creates for the admin SVM. If the cluster administrator also creates the scope for that cluster FPolicy policy, the SVM administrator cannot create a scope for that same policy. However, if the cluster administrator does not create a scope for the cluster FPolicy policy, then any SVM administrator can create the scope for that cluster policy. In the event that the SVM administrator creates a scope for that cluster FPolicy policy, the cluster administrator cannot subsequently create a cluster scope for that same cluster policy. This is because the cluster administrator cannot override the scope for the same cluster policy.

What the scope rules of precedence are

The following rules of precedence apply to scope configurations:

- When a share is included in the `-shares-to-include` parameter and the parent volume of the share is included in the `-volumes-to-exclude` parameter, `-volumes-to-exclude` has precedence over `-shares-to-include`.
- When an export policy is included in the `-export-policies-to-include` parameter and the parent volume of the export policy is included in the `-volumes-to-exclude` parameter, `-volumes-to-exclude` has precedence over `-export-policies-to-include`.
- An administrator can specify both `-file-extensions-to-include` and `-file-extensions-to-exclude` lists. The `-file-extensions-to-exclude` parameter is checked first before the `-file-extensions-to-include` parameter is checked.

What the FPolicy scope configuration contains

You can use the following list of available FPolicy scope configuration parameters to help you plan your configuration:

Note: When configuring what shares, export policies, volumes, and file extensions to include or exclude from the scope, the include and exclude parameters can contain regular expressions and can include metacharacters such as “?” and “*”.

Type of information	Option
SVM Specifies the SVM name on which you want to create an FPolicy scope. Each FPolicy configuration is defined within a single SVM. The external engine, policy event, policy scope, and policy that combine together to create an FPolicy policy configuration must all be associated with the same SVM.	`-vserver` *`vserver_name`*
Policy name Specifies the name of the FPolicy policy to which you want to attach the scope. The FPolicy policy must already exist.	`-policy-name` *`policy_name`*
Shares to include Specifies a comma-delimited list of shares to monitor for the FPolicy policy to which the scope is applied.	`-shares-to-include` *`share_name`*, ...
Shares to exclude Specifies a comma-delimited list of shares to exclude from monitoring for the FPolicy policy to which the scope is applied.	`-shares-to-exclude` *`share_name`*, ...
Volumes to include Specifies a comma-delimited list of volumes to monitor for the FPolicy policy to which the scope is applied.	`-volumes-to-include` *`volume_name`*, ...

Type of information	Option
Volumes to exclude Specifies a comma-delimited list of volumes to exclude from monitoring for the FPolicy policy to which the scope is applied.	`-volumes-to-exclude` *`volume_name`*, ...
Export policies to include Specifies a comma-delimited list of export policies to monitor for the FPolicy policy to which the scope is applied.	`-export-policies-to-include` *`export_policy_name`*, ...
Export policies to exclude Specifies a comma-delimited list of export policies to exclude from monitoring for the FPolicy policy to which the scope is applied.	`-export-policies-to-exclude` *`export_policy_name`*, ...
File extensions to include Specifies a comma-delimited list of file extensions to monitor for the FPolicy policy to which the scope is applied.	`-file-extensions-to-include` *`file_extensions`*, ...
File extension to exclude Specifies a comma-delimited list of file extensions to exclude from monitoring for the FPolicy policy to which the scope is applied.	`-file-extensions-to-exclude` *`file_extensions`*, ...
Is file extension check on directory enabled Specifies whether the file name extension checks apply to directory objects as well. If this parameter is set to `true`, the directory objects are subjected to the same extension checks as regular files. If this parameter is set to `false`, the directory names are not matched for extensions and notifications are sent for directories even if their name extensions do not match.	`-is-file-extension-check-on-directories-enabled` {`true`|`false`|}

Completing the FPolicy scope worksheet

You can use this worksheet to record the values that you need during the FPolicy scope configuration process. If a parameter value is required, you need to determine what value to use for those parameters before you configure the FPolicy scope.

You should record whether you want to include each parameter setting in the FPolicy scope configuration and then record the value for the parameters that you want to include.

Type of information	Required	Include	Your values
Storage Virtual Machine (SVM) name	Yes	Yes	
Policy name	Yes	Yes	

Type of information	Required	Include	Your values
Shares to include	No		
Shares to exclude	No		
Volumes to include	No		
Volumes to exclude	No		
Export policies to include	No		
Export policies to exclude	No		
File extensions to include	No		
File extension to exclude	No		
Is file extension check on directory enabled	No		

Creating the FPolicy configuration

There are several steps you must perform to creating an FPolicy configuration. First, you must plan your configuration. Then, you create an FPolicy external engine, an FPolicy event, and an FPolicy policy. You then create an FPolicy scope and attach it to the FPolicy policy, and then enable the FPolicy policy.

FPolicy is supported on Storage Virtual Machines (SVMs) with FlexVol volumes. FPolicy is not supported on SVMs with Infinite Volume.

Steps

1. *Creating the FPolicy external engine* on page 167
   The first step to creating an FPolicy configuration is to create an external engine. The external engine defines how FPolicy makes and manages connections to external FPolicy servers. If your configuration uses the native external engine for simple file blocking, you do not need to configure an external engine.
2. *Creating the FPolicy policy event* on page 167
   As part of creating an FPolicy policy configuration, you need to create an Fpolicy event. You associate the event with the FPolicy policy when it is created. An event defines which protocol to monitor and which file access events to monitor and filter.
3. *Creating the FPolicy policy* on page 168
   After creating an FPolicy external engine and FPolicy events, you create the FPolicy policy. The policy associates an external engine and one or more events to the policy. The FPolicy policy also specifies whether mandatory screening is required and whether the external FPolicy servers (FPolicy servers) have privileged access to data on the Storage Virtual Machine (SVM).

4. *Creating the FPolicy policy scope* on page 168
   After creating the FPolicy policy, you need to create an FPolicy scope. When creating the scope, you associate the scope with an FPolicy policy. A scope defines the boundaries on which the FPolicy policy applies. Scopes can include or exclude files based on shares, export policies, volumes, and file extensions.
5. *Enabling the FPolicy policy* on page 169
   After you are through configuring an FPolicy policy configuration, you enable the FPolicy policy. Enabling the policy sets its priority and starts file access monitoring for the policy.

Creating the FPolicy external engine

The first step to creating an FPolicy configuration is to create an external engine. The external engine defines how FPolicy makes and manages connections to external FPolicy servers. If your configuration uses the native external engine for simple file blocking, you do not need to configure an external engine.

Before you begin

The external engine worksheet should be completed.

Steps

1. Create the FPolicy external engine:

   vserver fpolicy policy external-engine create -vserver-name *vserver_name* -engine-name *engine_name* -primary-servers *IP_address*,... -port *integer* -ssl-option {no-auth|server-auth|mutual-auth} optional_parameters

2. Verify the FPolicy external engine configuration:

   vserver fpolicy policy external-engine show -vserver *vserver_name*

Creating the FPolicy policy event

As part of creating an FPolicy policy configuration, you need to create an Fpolicy event. You associate the event with the FPolicy policy when it is created. An event defines which protocol to monitor and which file access events to monitor and filter.

Before you begin

The FPolicy event worksheet should be completed.

Steps

1. Create the FPolicy event by using the following command:

   vserver fpolicy policy event create -vserver-name *vserver_name* -event-name *event_name* optional_parameters

2. Verify the FPolicy event configuration:

```
vserver fpolicy policy event show -vserver vserver_name
```

Creating the FPolicy policy

After creating an FPolicy external engine and FPolicy events, you create the FPolicy policy. The policy associates an external engine and one or more events to the policy. The FPolicy policy also specifies whether mandatory screening is required and whether the external FPolicy servers (FPolicy servers) have privileged access to data on the Storage Virtual Machine (SVM).

Before you begin

- The FPolicy policy worksheet should be completed.
- If you plan on configuring the policy to use FPolicy servers, the external engine must exist.
- At least one FPolicy event that you plan on associating with the FPolicy policy must exist.

Steps

1. Create the FPolicy policy by entering the following command:

   ```
   vserver fpolicy policy create -vserver-name vserver_name -policy-name
   policy_name -events event_name,... -engine engine_name
   optional_parameters
   ```

2. Verify the FPolicy policy configuration:

   ```
   vserver fpolicy policy show -vserver vserver_name
   ```

Creating the FPolicy policy scope

After creating the FPolicy policy, you need to create an FPolicy scope. When creating the scope, you associate the scope with an FPolicy policy. A scope defines the boundaries on which the FPolicy policy applies. Scopes can include or exclude files based on shares, export policies, volumes, and file extensions.

Before you begin

The FPolicy scope worksheet must be completed. The FPolicy policy must exist with an associated external engine (if the policy is configured to use external FPolicy servers) and must have at least one associated FPolicy event.

Steps

1. Create the FPolicy scope by entering the following command:

   ```
   vserver fpolicy policy scope create -vserver-name vserver_name -policy-
   name policy_name optional_parameters
   ```

2. Verify the FPolicy scope configuration:

   ```
   vserver fpolicy policy scope show -vserver vserver_name
   ```

Enabling the FPolicy policy

After you are through configuring an FPolicy policy configuration, you enable the FPolicy policy. Enabling the policy sets its priority and starts file access monitoring for the policy.

Before you begin

The FPolicy policy must exist with an associated external engine (if the policy is configured to use external FPolicy servers) and must have at least one associated FPolicy event. The FPolicy policy scope must exist and must be assigned to the FPolicy policy.

About this task

The priority is used when multiple policies are enabled on the Storage Virtual Machine (SVM) and more than one policy has subscribed to the same file access event. Policies that use the native engine configuration have a higher priority than policies for any other engine, regardless of the sequence number assigned to them when enabling the policy.

Note: A policy cannot be enabled on the admin SVM.

Steps

1. Enable the FPolicy policy by entering the following command:

   `vserver fpolicy enable -vserver-name` *`vserver_name`* `-policy-name` *`policy_name`* `-sequence-number` *`integer`*

2. Verify that the FPolicy policy is enabled:

   `vserver fpolicy show -vserver` *`vserver_name`*

Modifying FPolicy configurations

You can modify FPolicy configurations by modifying the elements that make up the configuration. You can modify external engines, FPolicy events, FPolicy scopes, and FPolicy policies. You can also enable or disable FPolicy policies. When you disable the FPolicy policy, file monitoring is discontinued for that policy.

It is recommended to disable the FPolicy policy before modifying the configuration.

Commands for modifying FPolicy configurations

You can modify FPolicy external engines, events, scopes, and policies.

If you want to modify...	Use this command...
External engines	`vserver fpolicy policy external-engine modify`

If you want to modify...	Use this command...
Events	`vserver fpolicy policy event modify`
Scopes	`vserver fpolicy policy scope modify`
Policies	`vserver fpolicy policy modify`

See the man pages for the commands for more information.

Enabling or disabling FPolicy policies

You can enable FPolicy policies after the configuration is complete. Enabling the policy sets its priority and starts file access monitoring for the policy. You can disable FPolicy policies if you want to stop file access monitoring for the policy.

Before you begin

Before enabling FPolicy policies, the FPolicy configuration must be completed.

About this task

- The priority is used when multiple policies are enabled on the Storage Virtual Machine (SVM) and more than one policy has subscribed to the same file access event.
- Policies that use the native engine configuration have a higher priority than policies for any other engine, regardless of the sequence number assigned to them when enabling the policy.
- If you want to change the priority of an FPolicy policy, you must disable the policy and then reenable it using the new sequence number.

Step

1. Perform the appropriate action:

If you want to...	Enter the following command...
Enable an FPolicy policy	**`vserver fpolicy enable -vserver-name`** ***`vserver_name`*** **`-policy-name`** ***`policy_name`*** **`-sequence-number`** ***`integer`***
Disable an FPolicy policy	**`vserver fpolicy disable -vserver-name`** ***`vserver_name`*** **`-policy-name`** ***`policy_name`***

Displaying information about FPolicy configurations

You might want to display information about FPolicy configurations to determine whether the configuration for each Storage Virtual Machine (SVM) is correct or to verify that an FPolicy policy configuration is enabled. You can display information about FPolicy external engines, FPolicy events, FPolicy scopes, and FPolicy policies.

How the show commands work

It is helpful when displaying information about the FPolicy configuration to understand how the `show` commands work.

A `show` command without additional parameters displays information in a summary form. Additionally, every `show` command has the same two mutually exclusive optional parameters, `-instance` and `-fields`.

When you use the `-instance` parameter with a `show` command, the command output displays detailed information in a list format. In some cases, the detailed output can be lengthy and include more information than you need. You can use the `-fields` *`fieldname[,fieldname...]`* parameter to customize the output so that it displays information only for the fields you specify. You can identity which fields that you can specify by entering `?` after the `-fields` parameter.

Note: The output of a `show` command with the `-fields` parameter might display other relevant and necessary fields related to the requested fields.

Every `show` command has one or more optional parameters that filter that output and enable you to narrow the scope of information displayed in command output. You can identity which optional parameters are available for a command by entering `?` after the `show` command.

The `show` command supports UNIX-style patterns and wildcards to enable you to match multiple values in command-parameters arguments. For example, you can use the wildcard operator (*), the NOT operator (!), the OR operator (|), the range operator (integer...integer), the less-than operator (<), the greater-than operator (>), the less-than or equal to operator (<=), and the greater-than or equal to operator (>=) when specifying values.

For more information about using UNIX-style patterns and wildcards, see the "Using the Data ONTAP command-line interface" section of the *Clustered Data ONTAP System Administration Guide for SVM Administrators*.

Commands for displaying information about FPolicy configurations

You use the `fpolicy show` commands to display information about the FPolicy configuration, including information about FPolicy external engines, events, scopes, and policies.

If you want to display information about FPolicy...	Use this command...
External engines	`vserver fpolicy policy external-engine show`
Events	`vserver fpolicy policy event show`
Scopes	`vserver fpolicy policy scope show`
Policies	`vserver fpolicy policy show`

See the man pages for the commands for more information.

Displaying information about FPolicy policy status

You can display information about the status for FPolicy policies to determine whether a policy is enabled, what external engine it is configured to use, what the sequence number is for the policy, and to which Storage Virtual Machine (SVM) the FPolicy policy is associated.

About this task

If you do not specify any parameter, the command displays the following information:

- SVM name
- Policy name
- Policy sequence number
- Policy status

In addition to displaying information about policy status for FPolicy policies configured on the cluster or a specific SVM, you can use command parameters to filter the command's output by other criteria.

You can specify the `-instance` parameter to display detailed information about listed policies. Alternatively, you can use the `-fields` parameter to display only the indicated fields in the command output, or `-fields ?` to determine what fields you can use.

Step

1. Display filtered information about FPolicy policy status by using the appropriate command:

If you want to display status information about policies...	Enter the command...
On the cluster	**vserver fpolicy show**
That have the specified status	**vserver fpolicy show -status {on|off}**
On a specified SVM	**vserver fpolicy show -vserver *vserver_name***
With the specified policy name	**vserver fpolicy show -policy-name *policy_name***
With the specified sequence number	**vserver fpolicy show -sequence-number *integer***
That use the specified external engine	**vserver fpolicy show -engine *engine_name***

The following example displays the information about FPolicy policies on the cluster:

```
cluster1::> vserver fpolicy show
  Vserver     Policy                 Sequence  Status Engine
                                     Number
  ----------- ---------------------- --------  ------ ---------
  FPolicy     cserver_policy                -  off    eng1
  vs1         v1p1                          -  off    eng2
  vs1         v1p2                          -  off    native
  vs1         v1p3                          -  off    native
  vs1         cserver_policy                -  off    eng1
  vs2         v1p1                          3  on     native
  vs2         v1p2                          1  on     eng3
  vs2         cserver_policy                2  on     eng1
```

Displaying information about enabled FPolicy policies

You can display information about enabled FPolicy policies to determine what FPolicy external engine it is configured to use, what the priority is for the policy, and to which Storage Virtual Machine (SVM) the FPolicy policy is associated.

About this task

If you do not specify any parameters, the command displays the following information:

- SVM name
- Policy name
- Policy priority

You can use command parameters to filter the command's output by specified criteria.

Step

1. Display information about enabled FPolicy policies by using the appropriate command:

If you want to display information about enabled policies...	Enter the command...
On the cluster	**vserver fpolicy show-enabled**
On a specified SVM	**vserver fpolicy show-enabled -vserver *vserver_name***
With the specified policy name	**vserver fpolicy show-enabled -policy-name *policy_name***
With the specified sequence number	**vserver fpolicy show-enabled -priority *integer***

The following example displays the information about enabled FPolicy policies on the cluster:

```
cluster1::> vserver fpolicy show-enabled
Vserver            Policy Name               Priority

------------------ ------------------------- ----------
vs1                pol_native                native
vs1                pol_native2               native
vs1                pol1                      2
vs1                pol2                      4
```

Managing FPolicy server connections

You can manage your FPolicy server connections by connecting to external FPolicy servers, disconnecting from external FPolicy servers, or displaying information about connections and connection status.

Connecting to external FPolicy servers

To enable file processing, you might need to manually connect to an external FPolicy server if the connection has previously been terminated. A connection is terminated after the server timeout is reached or due to some error. Alternatively, the administrator might manually terminate a connection.

About this task

If a fatal error occurs, the connection to the FPolicy server can be terminated. After resolving the issue that caused the fatal error, you must manually reconnect to the FPolicy server.

Steps

1. Connect to the external FPolicy server by using the `vserver fpolicy engine-connect` command.

   For more information about the command, see the man pages.

2. Verify that the external FPolicy server is connected by using the `vserver fpolicy show-engine` command.

   For more information about the command, see the man pages.

Disconnecting from external FPolicy servers

You might need to manually disconnect from an external FPolicy server. This might be desirable if the FPolicy server has issues with notification request processing or if you need to perform maintenance on the FPolicy server.

Steps

1. Disconnect from the external FPolicy server by using the `vserver fpolicy engine-disconnect` command.

   For more information about the command, see the man pages.

2. Verify that the external FPolicy server is disconnected by using the `vserver fpolicy show-engine` command.

   For more information about the command, see the man pages.

Displaying information about connections to external FPolicy servers

You can display status information about connections to external FPolicy servers (FPolicy servers) for the cluster or for a specified Storage Virtual Machine (SVM). This information can help you determine which FPolicy servers are connected.

About this task

If you do not specify any parameter, the command displays the following information:

- SVM name
- Node name
- FPolicy policy name
- FPolicy server IP address
- FPolicy server status
- FPolicy server type

In addition to displaying information about FPolicy connections on the cluster or a specific SVM, you can use command parameters to filter the command's output by other criteria.

You can specify the `-instance` parameter to display detailed information about listed policies. Alternatively, you can use the `-fields` parameter to display only the indicated fields in the command output. You can enter **`?`** after the `-fields` parameter to find out which fields you can use.

Step

1. Display filtered information about connection status between the node and the FPolicy server by using the appropriate command:

If you want to display connection status information about...	Enter the command...
FPolicy servers that you specify	**vserver fpolicy show-engine -server *IP_address***
FPolicy servers for a specified SVM	**vserver fpolicy show-engine -vserver *vserver_name***
FPolicy servers that are attached with a specified policy	**vserver fpolicy show-engine -policy-name *policy_name***
FPolicy servers with the server status that you specify	**vserver fpolicy show-engine -server-status *status*** The server status can be one of the following: • `connected` • `disconnected` • `connecting` • `disconnecting`
FPolicy servers with the specified type	**vserver fpolicy show-engine -server-type *type*** The FPolicy server type can be one of the following: • `primary` • `secondary`
FPolicy servers that were disconnected with the specified reason	**vserver fpolicy show-engine -disconnect-reason *text*** Disconnect can be due to multiple reasons. The following are common reasons for disconnect: • `Disconnect command received from CLI.` • `Error encountered while parsing notification response from FPolicy server.` • `FPolicy Handshake failed.` • `SSL handshake failed.` • `TCP Connection to FPolicy server failed.` • `The screen response message received from the FPolicy server is not valid.`

This example displays information about external engine connections to FPolicy servers on SVM vs1:

```
cluster1::> vserver fpolicy show-engine -vserver vs1
 FPolicy                                    Server-     Server-
 Vserver Policy    Node         Server       status      type
```

```
 ------- --------- ------------ ------------- ----------- ---------
 vs1     policy1   node1        1.1.1.1       connected   primary
```

This example displays information only about connected FPolicy servers:

```
cluster1::> vserver fpolicy show-engine -fields server -server-status
connected
 node       vserver policy-name server
 ---------- ------- ----------- -------
 node1      vs1     policy1     1.1.1.1
```

Glossary

To understand the file access and protocols management concepts in this document, you might need to know how certain terms are used.

A

ACL Access control list.

adapter A SCSI card, network card, hot-swap adapter, serial adapter, or VGA adapter that plugs into an expansion slot. Sometimes called *expansion card*.

address resolution The procedure for determining an address corresponding to the address of a LAN or WAN destination.

agent A process that gathers status and diagnostic information and forwards it to network management stations, for example, *SNMP agent*.

appliance A device that performs a single, well-defined function and is simple to install and operate.

ATM Asynchronous Transfer Mode. A network technology that combines the features of cell-switching and multiplexing to offer reliable and efficient network services. ATM provides an interface between devices such as workstations and routers, and the network.

AutoSupport A storage system daemon that triggers email messages from the customer site to technical support or another specified email recipient when there is a potential storage system problem.

B

big-endian A binary data format for storage and transmission in which the most significant byte comes first.

C

CIFS See *Common Internet File System (CIFS)*.

client A workstation or PC in a client-server architecture; that is, a computer system or process that requests services from and accepts the responses of another computer system or process.

cluster monitor The software that administers the relationship of nodes in a cluster.

community A logical relationship between an SNMP agent and one or more SNMP managers. A community is identified by name, and all members of the community have the same access privileges.

console The physical or virtual terminal that is used to monitor and control a storage system.

Copy-On-Write (COW) The technique for creating Snapshot copies without consuming excess disk space.

D

degraded mode The operating mode of a storage system when a disk in the RAID group fails or the batteries on the NVRAM card are low.

disk ID number The number assigned by the storage system to each disk when it probes the disks at startup.

disk shelf A shelf that contains disk drives and is attached to a storage system.

E

Ethernet adapter An Ethernet interface card.

expansion card A SCSI card, NVRAM card, network card, hot-swap card, or console card that plugs into a storage system expansion slot. Sometimes called an *adapter*.

expansion slot The slots on the storage system board into which you insert expansion cards.

F

FDDI adapter A Fiber Distributed Data Interface (FDDI) interface card.

FDDI-fiber An FDDI adapter that supports a fiber-optic cable.

FDDI-TP An FDDI adapter that supports a twisted-pair cable.

G

GID See *Group ID (GID)*.

Group ID (GID) The number used by UNIX systems to identify groups.

H

heartbeat A repeating signal transmitted from one storage system to the other that indicates that the storage system is in operation. Heartbeat information is also stored on disk.

hot spare disk A disk installed in the storage system that can be used to substitute for a failed disk. Before the disk failure, the hot spare disk is not part of the RAID disk array.

hot swap The process of adding, removing, or replacing a disk while the storage system is running.

hot swap adapter An expansion card that makes it possible to add or remove a hard disk with minimal interruption to file system activity.

I

inode	A data structure containing information about files on a storage system and in a UNIX file system.
interrupt switch	A switch on some storage system front panels used for debugging purposes.
L	
LAN Emulation (LANE)	The architecture, protocols, and services that create an Emulated LAN using ATM as an underlying network topology. LANE enables ATM-connected end systems to communicate with other LAN-based systems.
local storage system	The system you are logged in to.
M	
magic directory	A directory that can be accessed by name but does not show up in a directory listing. The .snapshot directories, except for the one at the mount point or at the root of the share, are magic directories.
mail host	The client host responsible for sending automatic email to technical support when certain storage system events occur.
Maintenance mode	An option when booting a storage system from a system boot disk. Maintenance mode provides special commands for troubleshooting hardware and configuration.
MIB	Management Information Base. ASCII files that describe the information that the SNMP agent sends to network management stations.
MIME	Multipurpose Internet Mail Extensions. A specification that defines the mechanisms for specifying and describing the format of Internet message bodies. An HTTP response containing the MIME Content-Type header allows the HTTP client to invoke the application that is appropriate for the data received.
N	
NDMP	Network Data Management Protocol. A protocol that allows storage systems to communicate with backup applications and provides capabilities for controlling the robotics of multiple tape backup devices.
network adapter	An Ethernet, FDDI, or ATM card.
network management station	See *NMS*.
NMS	Network Management Station. A host on a network that uses third-party network management application (SNMP manager) to process status and diagnostic information about a storage system.
null user	The Windows NT machine account used by applications to access remote data.

NVRAM cache Nonvolatile RAM in a storage system, used for logging incoming write data and NFS requests. Improves system performance and prevents loss of data in case of a storage system or power failure.

NVRAM card An adapter that contains the storage system's NVRAM cache.

NVRAM mirror A synchronously updated copy of the contents of the storage system NVRAM (nonvolatile random access memory) contents kept on the partner storage system.

P

panic A serious error condition causing the storage system or gateway to halt. Similar to a software crash in the Windows system environment.

parity disk The disk on which parity information is stored for a RAID4 disk drive array. In RAID groups using RAID-DP protection, two parity disks store the parity and double-parity information. Used to reconstruct data in failed disk blocks or on a failed disk.

PCI Peripheral Component Interconnect. The bus architecture used in newer storage system models.

POST Power-on self-tests. The tests run by a storage system after the power is turned on.

PVC Permanent Virtual Circuit. A link with a static route defined in advance, usually by manual setup.

Q

qtree A special subdirectory of the root of a volume that acts as a virtual subvolume with special attributes.

R

RAID Redundant Array of Independent Disks. A technique that protects against disk failure by computing parity information based on the contents of all the disks in an array. Storage systems use either RAID4, which stores all parity information on a single disk, or RAID-DP, which stores all parity information on two disks.

RAID disk scrubbing The process in which a system reads each disk in the RAID group and tries to fix media errors by rewriting the data to another disk area.

S

SCSI adapter An expansion card that supports SCSI disk drives and tape drives.

SCSI address The full address of a disk, consisting of the disk's SCSI adapter number and the disk's SCSI ID, such as 9a.1.

SCSI ID The number of a disk drive on a SCSI chain (0 to 6).

serial adapter An expansion card for attaching a terminal as the console on some storage system models.

serial console An ASCII or ANSI terminal attached to a storage system's serial port. Used to monitor and manage storage system operations.

share A directory or directory structure that has been made available to network users and can be mapped to a drive letter on a CIFS client. Also known as a *CIFS share*.

SID Security identifier used by the Windows operating system.

Snapshot copy An online, read-only copy of an entire file system that protects against accidental deletions or modifications of files without duplicating file contents. Snapshot copies enable users to restore files and to back up the storage system to tape while the storage system is in use.

SVC Switched Virtual Circuit. A connection established through signaling. The user defines the endpoints when the call is initiated.

system board A printed circuit board that contains a storage system's CPU, expansion bus slots, and system memory.

T

trap An asynchronous, unsolicited message sent by an SNMP agent to an SNMP manager indicating that an event has occurred on the storage system.

tree quota A type of disk quota that restricts the disk usage of a directory created by the quota qtree command. Different from user and group quotas that restrict disk usage by files with a given UID or GID.

U

UID user identification number.

Unicode A 16-bit character set standard. It was designed and is maintained by the nonprofit consortium Unicode Inc.

V

VCI Virtual Channel Identifier. A unique numerical tag defined by a 16-bit field in the ATM cell header that identifies a virtual channel over which the cell is to travel.

VGA adapter An expansion card for attaching a VGA terminal as the console.

volume

- For Data ONTAP, a logical entity that holds user data that is accessible through one or more of the supported access protocols, including Network File System (NFS), Common Internet File System (CIFS), Fibre Channel (FC), and Internet SCSI (iSCSI). The gateway treats an IBM volume as a disk.

- For IBM, the area on the storage array that is available for a gateway or non gateway host to read data from or write data to. The gateway documentation uses the term *array LUN* to describe this area.

VPI Virtual Path Identifier. An eight-bit field in the ATM cell header that indicates the virtual path over which the cell should be routed.

W

WAFL Write Anywhere File Layout. A file system designed for the storage system to optimize write performance.

WINS Windows Internet Name Service.

Copyright and trademark information

References in this documentation to IBM products, programs, or services do not imply that IBM intends to make these available in all countries in which IBM operates. Any reference to an IBM product, program, or service is not intended to state or imply that only IBM's product, program, or service may be used. Any functionally equivalent product, program, or service that does not infringe any of IBM's or NetApp's intellectual property rights may be used instead of the IBM or NetApp product, program, or service. Evaluation and verification of operation in conjunction with other products, except those expressly designated by IBM and NetApp, are the user's responsibility.

No part of this document covered by copyright may be reproduced in any form or by any means— graphic, electronic, or mechanical, including photocopying, recording, taping, or storage in an electronic retrieval system—without prior written permission of the copyright owner.

Software derived from copyrighted NetApp material is subject to the following license and disclaimer:

THIS SOFTWARE IS PROVIDED BY NETAPP "AS IS" AND WITHOUT ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE, WHICH ARE HEREBY DISCLAIMED. IN NO EVENT SHALL NETAPP BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT

(INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

NetApp reserves the right to change any products described herein at any time, and without notice. NetApp assumes no responsibility or liability arising from the use of products described herein, except as expressly agreed to in writing by NetApp. The use or purchase of this product does not convey a license under any patent rights, trademark rights, or any other intellectual property rights of NetApp.

The product described in this manual may be protected by one or more U.S.A. patents, foreign patents, or pending applications.

RESTRICTED RIGHTS LEGEND: Use, duplication, or disclosure by the government is subject to restrictions as set forth in subparagraph (c)(1)(ii) of the Rights in Technical Data and Computer Software clause at DFARS 252.277-7103 (October 1988) and FAR 52-227-19 (June 1987).

Trademark information

IBM, the IBM logo, and ibm.com are trademarks or registered trademarks of International Business Machines Corporation in the United States, other countries, or both. A complete and current list of other IBM trademarks is available on the Web at http://www.ibm.com/legal/copytrade.shtml

Linux is a registered trademark of Linus Torvalds in the United States, other countries, or both.

Microsoft, Windows, Windows NT, and the Windows logo are trademarks of Microsoft Corporation in the United States, other countries, or both.

UNIX is a registered trademark of The Open Group in the United States and other countries.

NetApp, the NetApp logo, Network Appliance, the Network Appliance logo, Akorri, ApplianceWatch, ASUP, AutoSupport, BalancePoint, BalancePoint Predictor, Bycast, Campaign Express, ComplianceClock, Cryptainer, CryptoShred, CyberSnap, Data Center Fitness, Data ONTAP, DataFabric, DataFort, Decru, Decru DataFort, DenseStak, Engenio, Engenio logo, E-Stack, ExpressPod, FAServer, FastStak, FilerView, Flash Accel, Flash Cache, Flash Pool, FlashRay, FlexCache, FlexClone, FlexPod, FlexScale, FlexShare, FlexSuite, FlexVol, FPolicy, GetSuccessful, gFiler, Go further, faster, Imagine Virtually Anything, Lifetime Key Management, LockVault, Mars, Manage ONTAP, MetroCluster, MultiStore, NearStore, NetCache, NOW (NetApp on the Web), Onaro, OnCommand, ONTAPI, OpenKey, PerformanceStak, RAID-DP,

ReplicatorX, SANscreen, SANshare, SANtricity, SecureAdmin, SecureShare, Select, Service Builder, Shadow Tape, Simplicity, Simulate ONTAP, SnapCopy, Snap Creator, SnapDirector, SnapDrive, SnapFilter, SnapIntegrator, SnapLock, SnapManager, SnapMigrator, SnapMirror, SnapMover, SnapProtect, SnapRestore, Snapshot, SnapSuite, SnapValidator, SnapVault, StorageGRID, StoreVault, the StoreVault logo, SyncMirror, Tech OnTap, The evolution of storage, Topio, VelocityStak, vFiler, VFM, Virtual File Manager, VPolicy, WAFL, Web Filer, and XBB are trademarks or registered trademarks of NetApp, Inc. in the United States, other countries, or both.

All other brands or products are trademarks or registered trademarks of their respective holders and should be treated as such.

NetApp, Inc. is a licensee of the CompactFlash and CF Logo trademarks.

NetApp, Inc. NetCache is certified RealSystem compatible.

Notices

This information was developed for products and services offered in the U.S.A.

IBM may not offer the products, services, or features discussed in this document in other countries. Consult your local IBM representative for information on the products and services currently available in your area. Any reference to an IBM product, program, or service is not intended to state or imply that only that IBM product, program, or service may be used. Any functionally equivalent product, program, or service that does not infringe on any IBM intellectual property right may be used instead. However, it is the user's responsibility to evaluate and verify the operation of any non-IBM product, program, or service.

IBM may have patents or pending patent applications covering subject matter described in this document. The furnishing of this document does not give you any license to these patents. You can send license inquiries, in writing to:

IBM Director of Licensing
IBM Corporation
North Castle Drive
Armonk, N.Y. 10504-1785
U.S.A.

For additional information, visit the web at:
http://www.ibm.com/ibm/licensing/contact/

The following paragraph does not apply to the United Kingdom or any other country where such provisions are inconsistent with local law:

INTERNATIONAL BUSINESS MACHINES CORPORATION PROVIDES THIS PUBLICATION "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Some states do not allow disclaimer of express or implied warranties in certain transactions, therefore, this statement may not apply to you.

This information could include technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes will be incorporated in new editions of the publication. IBM may make improvements and/or changes in the product(s) and/or the program(s) described in this publication at any time without notice.

Any references in this information to non-IBM web sites are provided for convenience only and do not in any manner serve as an endorsement of those web sites. The materials at those web sites are not part of the materials for this IBM product and use of those web sites is at your own risk.

IBM may use or distribute any of the information you supply in any way it believes appropriate without incurring any obligation to you.

Any performance data contained herein was determined in a controlled environment. Therefore, the results obtained in other operating environments may vary significantly. Some measurements may have been made on development-level systems and there is no guarantee that these measurements will be the same on generally available systems. Furthermore, some measurement may have been estimated through extrapolation. Actual results may vary. Users of this document should verify the applicable data for their specific environment.

Information concerning non-IBM products was obtained from the suppliers of those products, their published announcements or other publicly available sources. IBM has not tested those products and cannot confirm the accuracy of performance, compatibility or any other claims related to non-IBM products. Questions on the capabilities of non-IBM products should be addressed to the suppliers of those products.

If you are viewing this information in softcopy, the photographs and color illustrations may not appear.

Index

B

C

D

E

F

G

H

I

J

K

L

M

N

O

P

Q

R

S

T

U

V

W

X

NA 210-06380_A0, Printed in USA

SC27-6604-00