ONTAP® 9

SMB/CIFS and NFS Auditing and Security Tracing Guide

February 2019 | 215-11159_H0
doccomments@netapp.com

Updated for ONTAP 9.5

Contents

Deciding whether to use the SMB/CIFS and NFS Auditing and Security Tracing Guide

This guide describes the file access auditing features available for the SMB/CIFS and NFS protocols with ONTAP: native auditing and file policy management using FPolicy. It includes a conceptual overview, planning guidance, and detailed implementation instructions.

You should use this guide if you want to design and implement auditing of SMB/CIFS and NFS file access events under the following circumstances:

- Basic SMB/CIFS and NFS protocol file access has been configured.
- You want to create and maintain an auditing configuration using one of the following methods:
  - Native ONTAP functionality
  - External FPolicy servers

If you want to create a basic configuration using best practices, and you do not want a lot of conceptual background, you should choose among the following documentation:

- *SMB/CIFS Configuration Express Guide* (basic configuration using OnCommand System Manager)
  SMB/CIFS configuration express
- *SMB/CIFS and NFS Multiprotocol Configuration Express Guide* (basic configuration using OnCommand System Manager)
  SMB/CIFS and NFS multiprotocol express configuration
- *NFS Configuration Express Guide* (basic configuration using OnCommand System Manager)
  NFS express configuration
- *NFS Configuration Power Guide* (advanced configuration using the CLI)
  NFS configuration

If you want general information about SMB/CIFS and NFS protocol support in ONTAP, you should choose among the following documentation:

- *SMB/CIFS management*
- *NFS management*

If you require additional configuration or conceptual information, you should choose among the following documentation:

- Networking concepts and detailed implementation procedures
  - *Network and LIF management*
- Hyper-V and SQL Server configuration and management over the SMB protocol
  - *SMB/CIFS configuration for Microsoft Hyper-V and SQL Server*
- Automation of management tasks
  - *NetApp Documentation: OnCommand Workflow Automation (current releases)*
    OnCommand Workflow Automation enables you to run prepackaged workflows that automate management tasks such as the workflows described in Express and Power Guides.
- Technical Reports (TRs), which include additional information about ONTAP technology and interaction with external services

- *NetApp Technical Report 4067: NFS Best Practice and Implementation Guide*
- *NetApp Technical Report 4189: Clustered Data ONTAP CIFS Auditing Quick Start Guide*
- *NetApp Technical Report 4191: Best Practices Guide for Clustered Data ONTAP 8.2 Windows File Services*
- *NetApp Technical Report 4479: FPolicy Solution Guide for Clustered Data ONTAP: Northern Storage Suite (NSS)*

Auditing NAS events on SVMs

Auditing for NAS events is a security measure that enables you to track and log certain CIFS and NFS events on storage virtual machines (SVMs). This helps you track potential security problems and provides evidence of any security breaches. You can also stage and audit Active Directory central access policies to see what the result of implementing them would be.

CIFS events

You can audit the following events:

- SMB file and folder access events
  You can audit SMB file and folder access events on objects stored on FlexVol volumes belonging to the auditing-enabled SVMs.
- CIFS logon and logoff events
  You can audit CIFS logon and logoff events for CIFS servers on SVMs.
- Central access policy staging events
  You can audit the effective access of objects on CIFS servers using permissions applied through proposed central access policies. Auditing through the staging of central access policies enables you to see what the effects are of central access policies before they are deployed.
  Auditing of central access policy staging is set up using Active Directory GPOs; however, the SVM auditing configuration must be configured to audit central access policy staging events.
  Although you can enable central access policy staging in the auditing configuration without enabling Dynamic Access Control on the CIFS server, central access policy staging events are generated only if Dynamic Access Control is enabled. Dynamic Access Control is enabled through a CIFS server option. It is not enabled by default.

NFS events

You can audit file and directory NFSv4 access events on objects stored on SVMs.

Related concepts

SMB events that can be audited on page 13

How auditing works

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

Basic auditing concepts

To understand auditing in 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 ONTAP to store staging files. There is one staging volume per aggregate. Staging volumes are shared by all audit-enabled storage virtual machines (SVMs) to store audit records of data access for data volumes in that particular aggregate. Each SVM's audit records are stored in a separate directory within the staging volume.

Cluster administrators can view information about staging volumes, but most other volume operations are not permitted. Only ONTAP can create staging volumes. ONTAP automatically assigns a name to staging volumes. All staging volume names begin with `MDV_aud_` followed by the UUID of the aggregate containing that staging volume (for example: `MDV_aud_1d0131843d4811e296fc123478563412`.)

System volumes

A FlexVol volume that contains special metadata, such as metadata for file services audit logs. The admin SVM owns system volumes, which are visible across the cluster. Staging volumes are a type of system volume.

Consolidation task

A task that gets created when auditing is enabled. This long-running task on each SVM takes the audit records from staging files across the member nodes of the SVM. This task merges the audit records 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 ONTAP auditing process works

The ONTAP auditing process is different from the Microsoft auditing process. Before you configure auditing, you should understand how the 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. When the storage administrator enables auditing on the SVM, the auditing subsystem checks whether staging volumes are present. A staging volume must exist for each aggregate that contains 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 `The specified path "`*`/path`*`" does not exist in the namespace belonging to Vserver "`*`Vserver_name`*`"` error.
  Configuration creation fails if the directory exists but contains symlinks.
- Auditing schedules the consolidation task.

After this task is scheduled, 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 consolidation task verifies that all of the remaining logs are consolidated.

Guaranteed auditing

By default, auditing is guaranteed. 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 be completed 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 storage failover (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. 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 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 then 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 a 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.

The auditing consolidation job, which gets created when auditing is enabled, monitors the consolidation task and re-creates it if the consolidation task exits because of an error. Previously, users could delete the auditing consolidation job by using job manager commands such as `job delete`. Users are no longer allowed to delete the auditing consolidation job.

Related concepts

Basic auditing concepts on page 8
What the supported audit event log formats are on page 12
SMB events that can be audited on page 13

Related tasks

Creating a file and directory auditing configuration on SVMs on page 23

Related references

NFS file and directory access events that can be audited on page 18

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.

Related concepts

Troubleshooting auditing and staging volume space issues on page 44

Auditing requirements and considerations

Before you configure and enable auditing on your storage virtual machine (SVM), you need to be aware of certain requirements and considerations.

- The maximum number of auditing-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 auditing ACEs.
  When converting ACLs to mode bits, auditing ACEs are skipped. When converting mode bits to ACLs, auditing 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 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 event 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 event logs in the volume.

- Although you can enable central access policy staging in the auditing configuration without enabling Dynamic Access Control on the CIFS server, Dynamic Access Control must be enabled to generate central access policy staging events.

  Dynamic Access Control is not enabled by default.

Related concepts

Planning the auditing configuration on page 19

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, ONTAP converts the binary logs to the `EVTX` file format.

Related concepts

Viewing audit event logs on page 12
How active audit logs are viewed using Event Viewer on page 13

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 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.

Related concepts

How the ONTAP auditing process works on page 9
How active audit logs are viewed using Event Viewer on page 13
Considerations when auditing symlinks and hard links on page 16
Considerations when auditing alternate NTFS data streams on page 17

Related tasks

Determining what the complete path to the audited object is on page 15
Manually rotating the audit event logs on page 40

How active audit logs are viewed using Event Viewer

If the audit consolidation process is running on the cluster, the consolidation process appends new records to the active audit log file for audit-enabled storage virtual machines (SVMs). This active audit log can be accessed and opened over an SMB share in Microsoft Event Viewer.

In addition to viewing existing audit records, Event Viewer has a refresh option that enables you to refresh the content in the console window. Whether the newly appended logs are viewable in Event Viewer depends on whether oplocks are enabled on the share used to access the active audit log.

Oplocks setting on the share	Behavior
Enabled	Event Viewer opens the log that contains events written to it up to that point in time. The refresh operation does not refresh the log with new events appended by the consolidation process.
Disabled	Event Viewer opens the log that contains events written to it up to that point in time. The refresh operation refreshes the log with new events appended by the consolidation process.

Note: This information is applicable only for `EVTX` event logs. `XML` event logs can be viewed through SMB in a browser or through NFS using any XML editor or viewer.

SMB events that can be audited

ONTAP can audit certain SMB events, including certain file and folder access events, certain logon and logoff events, and central access policy staging events. Knowing which access events can be audited is helpful when interpreting results from the event logs.

The following additional SMB events can be audited in ONTAP 9.2 and later:

Event ID (EVT/EVTX)	Event	Description	Category
4670	Object permissions were changed	OBJECT ACCESS: Permissions changed.	File Access
4907	Object auditing settings were changed	OBJECT ACCESS: Audit settings changed.	File Access
4913	Object Central Access Policy was changed	OBJECT ACCESS: CAP changed.	File Access

The following SMB events can be audited in ONTAP 9.0 and later:

Event ID (EVT/EVTX)	Event	Description	Category
540/4624	An account was successfully logged on	LOGON/LOGOFF: Network (CIFS) logon.	Logon and Logoff
529/4625	An account failed to log on	LOGON/LOGOFF: Unknown user name or bad password.	Logon and Logoff
530/4625	An account failed to log on	LOGON/LOGOFF: Account logon time restriction.	Logon and Logoff
531/4625	An account failed to log on	LOGON/LOGOFF: Account currently disabled.	Logon and Logoff
532/4625	An account failed to log on	LOGON/LOGOFF: User account has expired.	Logon and Logoff
533/4625	An account failed to log on	LOGON/LOGOFF: User cannot log on to this computer.	Logon and Logoff
534/4625	An account failed to log on	LOGON/LOGOFF: User not granted logon type here.	Logon and Logoff
535/4625	An account failed to log on	LOGON/LOGOFF: User's password has expired.	Logon and Logoff
537/4625	An account failed to log on	LOGON/LOGOFF: Logon failed for reasons other than above.	Logon and Logoff
539/4625	An account failed to log on	LOGON/LOGOFF: Account locked out.	Logon and Logoff
538/4634	An account was logged off	LOGON/LOGOFF: Local or network user logoff.	Logon and Logoff
560/4656	Open Object/ Create Object	OBJECT ACCESS: Object (file or directory) open.	File Access
563/4659	Open Object with the Intent to Delete	OBJECT ACCESS: A handle to an object (file or directory) was requested with the Intent to Delete.	File Access
564/4660	Delete Object	OBJECT ACCESS: Delete Object (file or directory). ONTAP generates this event when a Windows client attempts to delete the object (file or directory).	File Access
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, ONTAP audits only the first SMB read and first SMB write operation (success or failure) on an object. This prevents 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
NA/4664	Hard link	OBJECT ACCESS: An attempt was made to create a hard link.	File Access

Event ID (EVT/EVTX)	Event	Description	Category
NA/4818	Proposed central access policy does not grant the same access permissions as the current central access policy	OBJECT ACCESS: Central Access Policy Staging.	File Access
NA/NA Data ONTAP Event ID 9999	Rename Object	OBJECT ACCESS: Object renamed. This is an ONTAP event. It is not currently supported by Windows as a single event.	File Access
NA/NA Data ONTAP Event ID 9998	Unlink Object	OBJECT ACCESS: Object unlinked. This is an ONTAP event. It is not currently supported by Windows as a single event.	File Access

Additional information about Event 4656

The **`HandleID`** tag in the audit `XML` event contains the handle of the object (file or directory) accessed. The **`HandleID`** tag for the EVTX 4656 event contains different information depending on whether the open event is for creating a new object or for opening an existing object:

- If the open event is an open request to create a new object (file or directory), the **`HandleID`** tag in the audit XML event shows an empty **`HandleID`** (for example: **`<Data Name="HandleID">00000000000000;00;00000000;00000000</Data>`**).

  The **`HandleID`** is empty because the OPEN (for creating a new object) request gets audited before the actual object creation happens and before a handle exists. Subsequent audited events for the same object have the right object handle in the **`HandleID`** tag.

- If the open event is an open request to open an existing object, the audit event will have the assigned handle of that object in the **`HandleID`** tag (for example: **`<Data Name="HandleID">00000000000401;00;000000ea;00123ed4</Data>`**).

Related concepts

Configuring audit policies on NTFS security-style files and directories on page 26

Related tasks

Determining what the complete path to the audited object is on page 15

Determining what the complete path to the audited object is

The object path printed in the **`<ObjectName>`** tag for an audit record contains the name of the volume (in parentheses) and the relative path from the root of the containing volume. If you want to determine the complete path of the audited object, including the junction path, there are certain steps you must take.

Steps

1. Determine what the volume name and relative path to audited object is by looking at the **`<ObjectName>`** tag in the audit event.

   #### Example

   In this example, the volume name is “data1” and the relative path to the file is `/dir1/file.txt`:

<Data Name="ObjectName">`(data1);/dir1/file.txt` **</Data>**

2. Using the volume name determined in the previous step, determine what the junction path is for the volume containing the audited object:

   Example

   In this example, the volume name is “data1” and the junction path for the volume containing the audited object is `/data/data1`:

   `volume show -junction -volume data1`

```
                                Junction                          Junction
Vserver   Volume       Language Active    Junction Path      Path Source
--------- ------------ -------- --------  ----------------- -----------
vs1       data1        en_US.UTF-8
                                true      /data/data1        RW_volume
```

3. Determine the full path to the audited object by appending the relative path found in the **`<ObjectName>`** tag to the junction path for the volume.

   Example

   In this example, the junction path for the volume:

   `/data/data1/dir1/file.text`

Considerations when auditing symlinks and hard links

There are certain considerations you must keep in mind when auditing symlinks and hard links.

An audit record contains information about the object being audited including the path to the audited object, which is identified in the **`ObjectName`** tag. You should be aware of how paths for symlinks and hard links are recorded in the **`ObjectName`** tag.

Symlinks

A symlink is a file with a separate inode that contains a pointer to the location of a destination object, known as the target. When accessing an object through a symlink, ONTAP automatically interprets the symlink and follows the actual canonical protocol agnostic path to the target object in the volume.

In the following example output, there are two symlinks, both pointing to a file named `target.txt`. One of the symlinks is a relative symlink and one is an absolute symlink. If either of the symlinks are audited, the **`ObjectName`** tag in the audit event contains the path to the file `target.txt`:

```
[root@host1 audit]# ls -l
total 0
lrwxrwxrwx 1 user1 group1 37 Apr  2 10:09 softlink_fullpath.txt -> /data/
audit/target.txt
lrwxrwxrwx 1 user1 group1 10 Apr  2 09:54 softlink.txt -> target.txt
-rwxrwxrwx 1 user1 group1 16 Apr  2 10:05 target.txt
```

Hard links

A hard link is a directory entry that associates a name with an existing file on a file system. The hard link points to the inode location of the original file. Similar to how ONTAP interprets symlinks, ONTAP interprets the hard link and follows the actual canonical path to the target object in the volume. When access to a hard link object is audited, the audit event records this absolute canonical path in the **`ObjectName`** tag rather than the hard link path.

Considerations when auditing alternate NTFS data streams

There are certain considerations you must keep in mind when auditing files with NTFS alternate data streams.

The location of an object being audited is recorded in an event record using two tags, the **`ObjectName`** tag (the path) and the **`HandleID`** tag (the handle). To properly identify which stream requests are being logged, you must be aware of what ONTAP records in these fields for NTFS alternate data streams:

- EVTX ID: 4656 events (open and create audit events)
    - The path of the alternate data stream is recorded in the **`ObjectName`** tag.
    - The handle of the alternate data stream is recorded in the **`HandleID`** tag.
- EVTX ID: 4663 events (all other audit events, such as read, write, getattr, and so on)
    - The path of the base file, not the alternate data stream, is recorded in the **`ObjectName`** tag.
    - The handle of the alternate data stream is recorded in the **`HandleID`** tag.

Example

The following example illustrates how to identify EVTX ID: 4663 events for alternate data streams using the **`HandleID`** tag. Even though the **`ObjectName`** tag (path) recorded in the read audit event is to the base file path, the **`HandleID`** tag can be used to identify the event as an audit record for the alternate data stream.

Stream file names take the form `base_file_name:stream_name`. In this example, the `dir1` directory contains a base file with an alternate data stream having the following paths:

```
/dir1/file1.txt
/dir1/file1.txt:stream1
```

Note: The output in the following event example is truncated as indicated; the output does not display all of the available output tags for the events.

For an EVTX ID 4656 (open audit event), the audit record output for the alternate data stream records the alternate data stream name in the **`ObjectName`** tag:

```
- <Event>
- <System>
  <Provider Name="Netapp-Security-Auditing" />
  <EventID>4656</EventID>
  <EventName>Open Object</EventName>
  [...]
  </System>
- <EventData>
  [...]
  <Data Name="ObjectType">Stream</Data>
  <Data Name="HandleID">00000000000401;00;000001e4;00176767</Data>
  <Data Name="ObjectName">(data1);/dir1/file1.txt:stream1</
Data>
  [...]
  </EventData>
  </Event>
- <Event>
```

For an EVTX ID 4663 (read audit event), the audit record output for the same alternate data stream records the base file name in the **`ObjectName`** tag; however, the handle in the **`HandleID`** tag is the alternative data stream's handle and can be used to correlate this event with the alternative data stream:

```
- <Event>
- <System>
  <Provider Name="Netapp-Security-Auditing" />
  <EventID>4663</EventID>
  <EventName>Read Object</EventName>
  [...]
  </System>
- <EventData>
  [...]
  <Data Name="ObjectType">Stream</Data>
  <Data Name="HandleID">00000000000401;00;000001e4;00176767</Data>
  <Data Name="ObjectName">(data1);/dir1/file1.txt</Data>
  [...]
  </EventData>
  </Event>
- <Event>
```

NFS file and directory access events that can be audited

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.

Related tasks

Configuring auditing for UNIX security style files and directories on page 29
Determining what the complete path to the audited object is on page 15

Planning the auditing configuration

Before you configure auditing on storage virtual machines (SVMs), 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 three optional parameters that you can specify:

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. The path can be up to 864 characters in length and must have read-write permissions. If the path is not valid, the audit configuration command fails. If the SVM is an SVM disaster recovery source, the log destination path cannot be on the root volume. This is because root volume content is not replicated to the disaster recovery destination.	`-destination` *`text`*	Yes	Yes	

Type of information	Option	Required	Include	Your values
Categories of events to audit Specifies the categories of events to audit. The following event categories can be audited: • File access events (both SMB and NFSv4) • CIFS logon and logoff events • Central access policy staging events Central access policy staging events are a new advanced auditing event available starting with Windows 2012 Active Directory domains. Central access policy staging events log information about changes to central access policies configured in Active Directory. • File share category events • Audit policy change events • Local user account management events • Security group management events • Authorization policy change events The default is to audit file access and CIFS logon and logoff events. **Note:** Before you can specify **`cap-staging`** as an event category, a CIFS server must exist on the SVM. Although you can enable central access policy staging in the auditing configuration without enabling Dynamic Access Control on the CIFS server, central access policy staging events are generated only if Dynamic Access Control is enabled. Dynamic Access Control is enabled through a CIFS server option. It is not enabled by default.	`-events` {**`file-ops`**|**`cifs-logon-logoff`**|**`cap-staging`**|**`file-share`**|**`audit-policy-change`**|**`user-account`**|**`security-group`**|**`authorization-policy-change`**}	No		
Log file output format Determines the output format of the audit logs. The output format can be either ONTAP-specific `XML` or Microsoft Windows `EVTX` log format. By default, the output format is `EVTX`.	`-format` {**`xml`**|**`evtx`**}	No		

Type of information	Option	Required	Include	Your values
Log files rotation limit Determines how many audit log files to retain before rotating the oldest log file out. For example, if you enter a value of **`5`**, the last five log files are retained. A value of **`0`** indicates that all the log files are retained. The default value is 0.	`-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		
Log rotation schedule: Day of week Determines the daily (day of week) schedule for rotating audit logs. Valid values are **`Sunday`** through **`Saturday`**, 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.		

Related concepts

Configuring file and folder audit policies on page 26
Auditing requirements and considerations on page 11
What the supported audit event log formats are on page 12

Related tasks

Creating a file and directory auditing configuration on SVMs on page 23

Creating a file and directory auditing configuration on SVMs

Creating a file and directory auditing configuration on your storage virtual machine (SVM) 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 23
   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 25
   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 25
   After completing the auditing configuration, you should verify that auditing is configured properly and is enabled.

Related concepts

Planning the auditing configuration on page 19
How to configure NTFS audit policies using the ONTAP CLI on page 29
Managing auditing configurations on page 39

Related tasks

Configuring NTFS audit policies using the Windows Security tab on page 26
Configuring auditing for UNIX security style files and directories on page 29
Enabling and disabling auditing on SVMs on page 40
Deleting an auditing configuration on page 43
Manually rotating the audit event logs on page 40

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).

Before you begin

If you plan on creating an auditing configuration for central access policy staging, a CIFS server must exist on the SVM.

Notes:

- Although you can enable central access policy staging in the auditing configuration without enabling Dynamic Access Control on the CIFS server, central access policy staging events are generated only if Dynamic Access Control is enabled.
  Dynamic Access Control is enabled through a CIFS server option. It is not enabled by default.
- If the arguments of a field in a command is invalid, for example, invalid entries for fields, duplicate entries, and non-existent entries, then the command fails before the audit phase.

Such failures do not generate an audit record.

About this task

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

Step

1. Using the information in the planning worksheet, create the auditing configuration to rotate audit logs based on log size or a schedule:

If you want to rotate audit logs by...	Enter...
Log size	**vserver audit create -vserver** ***vserver_name*** **-destination** ***path*** **-events [{file-ops|cifs-logon-logoff|cap-staging|file-share|audit-policy-change|user-account|security-group|authorization-policy-change}] [-format {xml|evtx}] [-rotate-limit** ***integer*****] [-rotate-size {*****integer*****[KB|MB|GB|TB|PB]}]**
A schedule	**vserver audit create -vserver** ***vserver_name*** **-destination** ***path*** **-events [{file-ops|cifs-logon-logoff|cap-staging}] [-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 you are configuring time-based audit log rotation.

Examples

The following example creates an auditing configuration that audits file operations and CIFS logon and logoff events (the default) using size-based rotation. 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 auditing configuration that audits file operations and CIFS logon and logoff events (the default) using size-based rotation. The log format is `EVTX` (the default). The log file size limit is **`100 MB`** (the default), and the log rotation limit is **`5`**:

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

The following example creates an auditing configuration that audits file operations, CIFS logon and logoff events, and central access policy staging events 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. The log rotation limit is **`5`**:

```
cluster1::> vserver audit create -vserver vs1 -destination /audit_log -
events file-ops,cifs-logon-logoff,file-share,audit-policy-change,user-
account,security-group,authorization-policy-change,cap-staging -rotate-
schedule-month all -rotate-schedule-dayofweek all -rotate-schedule-hour 12 -
rotate-schedule-minute 30 -rotate-limit 5
```

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.

About this task

When an SVM disaster recovery ID discard configuration is first started (after the SnapMirror initialization is complete) and the SVM has an auditing configuration, ONTAP automatically disables the auditing configuration. Auditing is disabled on the read-onlySVM to prevent the staging volumes from filling up. You can enable auditing only after the SnapMirror relationship is broken and the SVM is read-write.

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 command displays in list form all auditing configuration information for storage virtual machine (SVM) vs1:

   vserver audit show -instance -vserver vs1

```
                             Vserver: vs1
                      Auditing state: true
                Log Destination Path: /audit_log
       Categories of Events to Audit: file-ops
                          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). 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 policies. SMB and NFS audit policies have different configuration requirements and audit capabilities.

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

Related concepts

How the ONTAP auditing process works on page 9
SMB events that can be audited on page 13
Displaying information about audit policies applied to files and directories on page 30

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 ONTAP CLI.

Configuring NTFS audit policies using the Windows Security tab

You can configure NTFS 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 policies on data residing on a Windows client, which enables you to use the same GUI interface that you 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 system access control lists (SACLs).

About this task

Configuring NTFS audit policies is done by adding entries to NTFS 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, SACLs for file and folder auditing, or both SACLs and DACLs.

To set NTFS audit policies using the Windows Security tab, 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** box:

a. Select a **Drive** letter.

b. In the **Folder** box, type the CIFS server name that contains the share, holding the data you want to audit and the name of the share.

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

   Example

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

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 the file or directory, and then 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 group 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 box setting defaults to **This object only**.

Note: Because 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.

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

11. If you do not want the auditing setting to propagate to subsequent files and folders of the original container, select the **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.

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

15. Click **OK**.

    The Auditing box closes.

Related concepts

SMB events that can be audited on page 13

Related tasks

Displaying information about audit policies using the Windows Security tab on page 30

How to configure NTFS audit policies using the ONTAP CLI

You can configure audit policies on files and folders using the 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 ONTAP command family. See the man pages for more information about using these commands to configure and add NTFS SACLs to files and folders.

Related concepts

SMB events that can be audited on page 13

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.

Related references

NFS file and directory access events that can be audited on page 18

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.

Related concepts

Configuring file and folder audit policies on page 26

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 information to validate your security configuration or to troubleshoot auditing issues.

About this task

You must provide the name of the storage virtual machine (SVM) and the path to the files or folders whose audit information you want to display. You can display the output in summary form or as a detailed list.

- NTFS security-style volumes and qtrees use only NTFS system access control lists (SACLs) for audit policies.
- Files and folders in a mixed security-style volume with NTFS effective security can have NTFS audit policies applied to them.

  Mixed security-style volumes and qtrees can contain some files and directories that use UNIX file permissions, either mode bits or NFSv4 ACLs, and some files and directories that use NTFS file permissions.
- The top level of a mixed security-style volume can have either UNIX or NTFS effective security and might or might not contain NTFS SACLs.
- Because Storage-Level Access Guard security can be configured on a mixed security-style volume or qtree even if the effective security style of the volume root or qtree is UNIX, the output for a volume or qtree path where Storage-Level Access Guard is configured might display both regular file and folder NFSv4 SACLs and Storage-Level Access Guard NTFS SACLs.

- If the path that is entered in the command is to data with NTFS effective security, the output also displays information about Dynamic Access Control ACEs if Dynamic Access Control is configured for the given file or directory path.
- When displaying security information about files and folders with NTFS effective security, UNIX-related output fields contain display-only UNIX file permission information.

  NTFS security-style files and folders use only NTFS file permissions and Windows users and groups when determining file access rights.
- ACL output is displayed only for files and folders with NTFS or NFSv4 security.

  This field is empty for files and folders using UNIX security that have only mode bit permissions applied (no NFSv4 ACLs).
- The owner and group output fields in the ACL output apply only in the case of NTFS security descriptors.

Step

1. Display file and directory audit policy settings with the desired level of detail:

If you want to display information...	Enter the following command...
In summary form	**`vserver security file-directory show -vserver`** ***`vserver_name`*** **`-path`** ***`path`***
As a detailed list	**`vserver security file-directory show -vserver`** ***`vserver_name`*** **`-path`** ***`path`*** **`-expand-mask true`**

Examples

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

```
cluster::> vserver security file-directory show -vserver vs1 -path /corp
                Vserver: vs1
              File Path: /corp
      File Inode Number: 357
         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
```

The following example displays the audit policy information for the path `/datavol1` in SVM vs1. The path contains both regular file and folder SACLs and Storage-Level Access Guard SACLs.

```
cluster::> vserver security file-directory show -vserver vs1 -path /datavol1

                Vserver: vs1
              File Path: /datavol1
        File Inode Number: 77
         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:0xaa14
                         Owner:BUILTIN\Administrators
                         Group:BUILTIN\Administrators
                         SACL - ACEs
                           AUDIT-EXAMPLE\marketing-0xf01ff-OI|CI|FA
                         DACL - ACEs
                           ALLOW-EXAMPLE\Domain Admins-0x1f01ff-OI|CI
                           ALLOW-EXAMPLE\marketing-0x1200a9-OI|CI

                         Storage-Level Access Guard security
                         SACL (Applies to Directories):
                           AUDIT-EXAMPLE\Domain Users-0x120089-FA
                           AUDIT-EXAMPLE\engineering-0x1f01ff-SA
                         DACL (Applies to Directories):
                           ALLOW-EXAMPLE\Domain Users-0x120089
                           ALLOW-EXAMPLE\engineering-0x1f01ff
                           ALLOW-NT AUTHORITY\SYSTEM-0x1f01ff
                         SACL (Applies to Files):
                           AUDIT-EXAMPLE\Domain Users-0x120089-FA
                           AUDIT-EXAMPLE\engineering-0x1f01ff-SA
                         DACL (Applies to Files):
                           ALLOW-EXAMPLE\Domain Users-0x120089
                           ALLOW-EXAMPLE\engineering-0x1f01ff
                           ALLOW-NT AUTHORITY\SYSTEM-0x1f01ff
```

Ways to display information about file security and audit policies

You can use the wildcard character (*) to display information about file security and audit policies of all files and directories under a given path or a root volume.

The wildcard character (*) can be used as the last subcomponent of a given directory path below which you want to display information of all files and directories. If you want to display information of a particular file or directory named as “*”, then you need to provide the complete path inside double quotes (“ ”).

Example

The following command with the wildcard character displays the information about all files and directories below the path `/1/` of SVM vs1:

```
cluster::> vserver security file-directory show -vserver vs1 –path /1/*

                    Vserver: vs1
                  File Path: /1/1
             Security Style: mixed
            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:0x8514
                             Owner:BUILTIN\Administrators
                             Group:BUILTIN\Administrators
                             DACL - ACEs
                             ALLOW-Everyone-0x1f01ff-OI|CI (Inherited)
                    Vserver: vs1
                  File Path: /1/1/abc
             Security Style: mixed
            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:0x8404
                             Owner:BUILTIN\Administrators
                             Group:BUILTIN\Administrators
                             DACL - ACEs
                             ALLOW-Everyone-0x1f01ff-OI|CI (Inherited)
```

The following command displays the information of a file named as "*" under the path `/vol1/a` of SVM vs1. The path is enclosed within double quotes (" ").

```
cluster::> vserver security file-directory show -vserver vs1 -path "/
vol1/a/*"

                 Vserver: vs1
               File Path: “/vol1/a/*”
          Security Style: mixed
         Effective Style: unix
          DOS Attributes: 10
  DOS Attributes in Text: ----D---
Expanded Dos Attributes: -
            Unix User Id: 1002
           Unix Group Id: 65533
          Unix Mode Bits: 755
  Unix Mode Bits in Text: rwxr-xr-x
                    ACLs: NFSV4 Security Descriptor
                          Control:0x8014
                          SACL - ACEs
                            AUDIT-EVERYONE@-0x1f01bf-FI|DI|SA|FA
                          DACL - ACEs
                            ALLOW-EVERYONE@-0x1f00a9-FI|DI
                            ALLOW-OWNER@-0x1f01ff-FI|DI
                            ALLOW-GROUP@-0x1200a9-IG
```

CLI change events that can be audited

ONTAP can audit certain CLI change events, including certain cifs-share events, certain audit policy events, certain local security group events, local user group events, and authorization policy events. Understanding which change events can be audited is helpful when interpreting results from the event logs.

You can manage storage virtual machine (SVM) auditing CLI change events by manually rotating the audit logs, enabling or disabling auditing, displaying information about auditing change events, modifying auditing change events, and deleting auditing change events.

As an administrator, if you execute any command to change configuration related to the cifs-share, local user-group, local security-group, authorization-policy, and audit-policy events, a record generates and the corresponding event gets audited:

Auditing Category	Events	Event IDs	Run this command...
Mhost Auditing	policy-change	[4719] Audit configuration changed	**`vserver audit disable|enable|modify`**
	file-share	[5142] Network share was added	**`vserver cifs share create`**

		[5143] Network share was modified	**`vserver cifs share modify`**
			`vserver cifs share create|modify|delete`
			`vserver cifs share add|remove`

		[5144] Network share deleted	**`vserver cifs share delete`**

Auditing Category	Events	Event IDs	Run this command...

Auditing	user-account	[4720] Local user created	**`vserver cifs users-and-groups local-user create`**
			`vserver services name-service unix-user create`

		[4722] Local user enabled	**`vserver cifs users-and-groups local-user create|modify`**
		[4724] Local user password reset	**`vserver cifs users-and-groups local-user set-password`**
		[4725] Local user disabled	**`vserver cifs users-and-groups local-user create|modify`**

		[4726] Local user deleted	**`vserver cifs users-and-groups local-user delete`**
			`vserver services name-service unix-user delete`

		[4738] Local user Change	**`vserver cifs users-and-groups local-user modify`**
			`vserver services name-service unix-user modify`

		[4781] Local user Rename	**`vserver cifs users-and-groups local-user rename`**

	security-group	[4731] Local Security Group created	**`vserver cifs users-and-groups local-group create`**
			`vserver services name-service unix-group create`

		[4734] Local Security Group deleted	**`vserver cifs users-and-groups local-group delete`**
			`vserver services name-service unix-group delete`

		[4735] Local Security Group Modified	**`vserver cifs users-and-groups local-group rename|modify`**
			`vserver services name-service unix-group modify`

		[4732] User added to Local Group	**`vserver cifs users-and-groups local-group add-members`**
			`vserver services name-service unix-group adduser`

		[4733] User Removed from Local Group	**`vserver cifs users-and-groups local-group remove-members`**
			`vserver services name-service unix-group deluser`

	authorization-policy-change	[4704] User Rights Assigned	**`vserver cifs users-and-groups privilege add-privilege`**
		[4705] User Rights Removed	**`vserver cifs users-and-groups privilege remove-privilege|reset-privilege`**

How to manage file-share event

When a file-share event is configured for a storage virtual machine (SVM) and an audit is enabled, audit events are generated. The file-share events are generated when the CIFS network share is modified using **`vserver cifs share`** related commands.

The file-share events with the event-ids 5142, 5143, and 5144 are generated when a CIFS network share is added, modified, or deleted for the SVM. The CIFS network share configuration is modified using the **`cifs share access control create|modify|delete`** commands.

The following example displays a file-share event with the ID 5143 is generated, when a share object called 'audit_dest' is created:

```
netapp-clus1::*> cifs share create -share-name audit_dest -path /audit_dest
- System
  - Provider
   [ Name]  NetApp-Security-Auditing
   [ Guid]  {3CB2A168-FE19-4A4E-BDAD-DCF422F13473}
   EventID 5142
   EventName Share Object Added
   ...
   ...
  ShareName audit_dest
  SharePath /audit_dest
  ShareProperties oplocks;browsable;changenotify;show-previous-versions;
  SD O:BAG:S-1-5-21-2447422786-1297661003-4197201688-513D:(A;;FA;;;WD)
```

How to manage audit-policy-change event

When an audit-policy-change event is configured for a storage virtual machine (SVM) and an audit is enabled, audit events are generated. The audit-policy-change events are generated when an audit policy is modified using **`vserver audit`** related commands.

The audit-policy-change event with the event-id 4719 is generated whenever an audit policy is disabled, enabled, or modified and helps to identify when a user attempts to disable auditing to cover the tracks. It is configured by default and requires diagnostic privilege to disable.

The following example displays an audit-policy change event with the ID 4719 generated, when an audit is disabled:

```
netapp-clus1::*> vserver audit disable -vserver vserver_1
- System
  - Provider
   [ Name]  NetApp-Security-Auditing
   [ Guid]  {3CB2A168-FE19-4A4E-BDAD-DCF422F13473}
   EventID 4719
   EventName Audit Disabled
   ...
   ...
  SubjectUserName admin
  SubjectUserSid 65533-1001
  SubjectDomainName ~
  SubjectIP console
  SubjectPort
```

How to manage user-account event

When a user-account event is configured for a storage virtual machine (SVM) and an audit is enabled, audit events are generated.

The user-account events with event-ids 4720, 4722, 4724, 4725, 4726, 4738, and 4781 are generated when a local CIFS or NFS user is created or deleted from the system, local user account is enabled, disabled or modified, and local CIFS user password is reset or changed. The user-account events are generated when a user account is modified using **`vserver cifs users-and-groups <local user>`** and **`vserver services name-service <unix user>`** commands.

The following example displays an user account event with the ID 4720 generated, when a local CIFS user is created:

```
netapp-clus1::*> vserver cifs users-and-groups local-user create -user-name
testuser -is-account-disabled false -vserver vserver_1
Enter the password:
Confirm the password:

- System
  - Provider
   [ Name]  NetApp-Security-Auditing
   [ Guid]  {3CB2A168-FE19-4A4E-BDAD-DCF422F13473}
   EventID 4720
   EventName Local Cifs User Created
   ...
   ...
  TargetUserName testuser
  TargetDomainName NETAPP-CLUS1
  TargetSid S-1-5-21-2447422786-1297661003-4197201688-1003
  TargetType CIFS
  DisplayName testuser
  PasswordLastSet 1472662216
  AccountExpires NO
  PrimaryGroupId 513
  UserAccountControl %%0200
  SidHistory ~
  PrivilegeList ~
```

The following example displays an user account event with the ID 4781 generated, when the local CIFS user created in the preceding example is renamed:

```
 netapp-clus1::*> vserver cifs users-and-groups local-user rename -user-name
testuser -new-user-name testuser1
- System
  - Provider
   [ Name]  NetApp-Security-Auditing
   [ Guid]  {3CB2A168-FE19-4A4E-BDAD-DCF422F13473}
   EventID 4781
   EventName Local Cifs User Renamed
   ...
   ...
  OldTargetUserName testuser
  NewTargetUserName testuser1
  TargetDomainName NETAPP-CLUS1
  TargetSid S-1-5-21-2447422786-1297661003-4197201688-1000
  TargetType CIFS
  SidHistory ~
  PrivilegeList ~
```

How to manage security-group event

When a security-group event is configured for a storage virtual machine (SVM) and an audit is enabled, audit events are generated.

The security-group events with event-ids 4731, 4732, 4733, 4734, and 4735 are generated when a local CIFS or NFS group is created or deleted from the system, and local user is added or removed from the group. The security-group-events are generated when a user account is modified using **`vserver cifs users-and-groups <local-group>`** and **`vserver services name-service <unix-group>`** commands.

The following example displays a security group event with the ID 4731 generated, when a local UNIX security group is created:

```
netapp-clus1::*> vserver services name-service unix-group create -name
testunixgroup -id 20
- System
  - Provider
   [ Name]  NetApp-Security-Auditing
   [ Guid]  {3CB2A168-FE19-4A4E-BDAD-DCF422F13473}
   EventID 4731
   EventName Local Unix Security Group Created
   ...
   ...
  SubjectUserName admin
  SubjectUserSid 65533-1001
  SubjectDomainName ~
  SubjectIP console
  SubjectPort
  TargetUserName testunixgroup
  TargetDomainName
  TargetGid 20
  TargetType NFS
  PrivilegeList ~
  GidHistory ~
```

How to manage authorization-policy-change event

When authorization-policy-change event is configured for a storage virtual machine (SVM) and an audit is enabled, audit events are generated.

The authorization-policy-change events with the event-ids 4704 and 4705 are generated whenever the authorization rights are granted or revoked for a CIFS user and CIFS group. The authorization-policy-change events are generated when the authorization rights are assigned or revoked using **`vserver cifs users-and-groups privilege`** related commands.

The following example displays an authorization policy event with the ID 4704 generated, when the authorization rights for a CIFS user group are assigned:

```
netapp-clus1::*> vserver cifs users-and-groups privilege add-privilege -user-
or-group-name testcifslocalgroup -privileges *
- System
  - Provider
   [ Name]  NetApp-Security-Auditing
   [ Guid]  {3CB2A168-FE19-4A4E-BDAD-DCF422F13473}
   EventID 4704
   EventName User Right Assigned
   ...
   ...
  TargetUserOrGroupName testcifslocalgroup
  TargetUserOrGroupDomainName NETAPP-CLUS1
  TargetUserOrGroupSid S-1-5-21-2447422786-1297661003-4197201688-1004;
  PrivilegeList
SeTcbPrivilege;SeBackupPrivilege;SeRestorePrivilege;SeTakeOwnershipPrivilege;
SeSecurityPrivilege;SeChangeNotifyPrivilege;
  TargetType CIFS
```

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.

Related concepts

Troubleshooting auditing and staging volume space issues on page 44

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 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.

Related concepts

Viewing audit event logs on page 12

Related tasks

Creating a file and directory auditing configuration on SVMs on page 23

Enabling and disabling auditing on SVMs

You can enable or disable auditing on storage virtual machines (SVMs). 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

Before you can enable auditing on the SVM, the SVM's auditing configuration must already exist.

About this task

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: vs1
                      Auditing state: true
```

```
                Log Destination Path: /audit_log
       Categories of Events to Audit: file-ops, cifs-logon-logoff
                          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: 10
```

The following example disables auditing for SVM vs1:

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

                             Vserver: vs1
                      Auditing state: false
                Log Destination Path: /audit_log
       Categories of Events to Audit: file-ops, cifs-logon-logoff
                          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: 10
```

Related tasks

Deleting an auditing configuration on page 43

Displaying information about auditing configurations

You can display information about auditing configurations. 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 categories of events to audit
- 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 a summary of the auditing configuration for all SVMs:

```
cluster1::> vserver audit show

 Vserver     State  Event Types Log Format Target Directory
 ----------- ------ ----------- ---------- --------------------
 vs1         false  file-ops    evtx       /audit_log
```

The following example displays, in list form, all auditing configuration information for all SVMs:

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

                             Vserver: vs1
                      Auditing state: true
                Log Destination Path: /audit_log
       Categories of Events to Audit: file-ops
                          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
```

Related tasks

Creating a file and directory auditing configuration on SVMs on page 23

Commands for modifying auditing configurations

If you want to change an auditing setting, you can modify the current configuration at any time, including modifying the log path destination and log format, modifying the categories of events to audit, how to automatically save log files, and specify the maximum number of log files to save.

If you want to...	Use this command...	For more information, see...
Modify the log destination path	`vserver audit modify` with the `-destination` parameter	
Modify the category of events to audit	`vserver audit modify` with the `-events` parameter **Note:** To audit central access policy staging events, the Dynamic Access Control (DAC) CIFS server option must be enabled on the storage virtual machine (SVM).	
Modify the log format	`vserver audit modify` with the `-format` parameter	
Enabling automatic saves based on internal log file size	`vserver audit modify` with the `-rotate-size` parameter	

If you want to...	Use this command...	For more information, see...
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	

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
   ```

Related tasks

Enabling and disabling auditing on SVMs on page 40

What the process is when reverting

If you plan to revert the cluster, you should be aware of the revert process ONTAP follows when there are auditing-enabled storage virtual machines (SVMs) in the cluster. You must take certain actions before reverting.

Reverting to a version of ONTAP that does not support the auditing of CIFS logon and logoff events and central access policy staging events

Support for auditing of CIFS logon and logoff events and for central access policy staging events starts with clustered Data ONTAP 8.3. If you are reverting to a version of ONTAP that does not support these event types and you have auditing configurations that monitor these event types, you must change the auditing configuration for those audit-enabled SVMs before reverting. You must modify the configuration so that only file-op events are audited.

Related tasks

Enabling and disabling auditing on SVMs on page 40

Deleting an auditing configuration on page 43

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.

Related concepts

Aggregate space considerations when enabling auditing on page 11

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 must 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.

  Note: Data access is denied in the following cases:

  - If the destination directory is deleted.
  - If the file limit on a volume, which hosts the destination directory, reaches to its maximum level.

For more information about how to view information about volumes and increasing volume size, see the *Logical Storage Management Guide*.

For more information about how to view information about aggregates and managing aggregates, see the *Disks and Aggregates Power Guide*.

Related information

ONTAP concepts
Logical Storage Management Guide
Disks and Aggregates Power Guide

How to troubleshoot space issues related to the staging volumes

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, you need 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 you 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 you can view information about staging volumes; SVM administrators cannot.

All staging volume names begin with `MDV_aud_` followed by the UUID of the aggregate containing that staging volume. The following example shows four system volumes on the admin SVM, which were automatically created when a file services auditing configuration was created for a data SVM in the cluster:

```
cluster1::> volume show -vserver cluster1
Vserver   Volume       Aggregate    State      Type       Size  Available Used%
--------- ------------ ------------ ---------- ---- ---------- ---------- -----
cluster1  MDV_aud_1d0131843d4811e296fc123478563412
                       aggr0        online     RW          2GB     1.90GB    5%
cluster1  MDV_aud_8be27f813d7311e296fc123478563412
                       root_vs0     online     RW          2GB     1.90GB    5%
cluster1  MDV_aud_9dc4ad503d7311e296fc123478563412
                       aggr1        online     RW          2GB     1.90GB    5%
cluster1  MDV_aud_a4b887ac3d7311e296fc123478563412
                       aggr2        online     RW          2GB     1.90GB    5%
4 entries were displayed.
```

If there is insufficient space in the staging volumes, you 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 you can increase the size of the volume. Only you can increase the size of an aggregate; SVM administrators cannot.

If one or more aggregates have an available space of less than 2 GB, the SVM audit creation fails. When the SVM audit creation fails, the staging volumes that were created are deleted.

Related information

ONTAP concepts

Using FPolicy for file monitoring and management on SVMs

FPolicy is a file access notification framework that is used to monitor and manage file access events on storage virtual machines (SVMs).

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

Note: FPolicy is not supported on SVMs 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. The ONTAP FPolicy framework manages activities on the cluster and sends notifications to external FPolicy servers. External FPolicy servers process notifications sent by ONTAP FPolicy.

The ONTAP framework creates and maintains the FPolicy configuration, monitors file events, and sends notifications to external FPolicy servers. 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.

Related concepts

Roles that cluster components play with FPolicy implementation on page 47
How FPolicy works with external FPolicy servers on page 48
How FPolicy services work across SVM namespaces on page 51
FPolicy configuration types on page 51
What the steps for setting up an FPolicy configuration are on page 55

What synchronous and asynchronous notifications 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 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.

Related concepts

How control channels are used for FPolicy communication on page 48
How privileged data access channels are used for synchronous communication on page 48

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). For example:

- File access and audit logging
- Storage resource management

Synchronous applications are ones where data access is altered or data is modified by the external FPolicy server. For example:

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

You can use the SDK for FPolicy to identify and implement other applications as well.

Roles that cluster components play with FPolicy implementation

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. After 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.
- Manages the passthrough-read data connection established by the FPolicy server for servicing client requests when passthrough-read is enabled.

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.

Related concepts

What granting super user credentials for privileged data access means on page 49

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

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 the following privileges when the FPolicy server accesses data:

- Avoid permission checks
  The user avoids checks on files and directory access.

- Special locking privileges
  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.

- Bypass 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.

Related concepts

Planning the FPolicy policy configuration on page 70

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

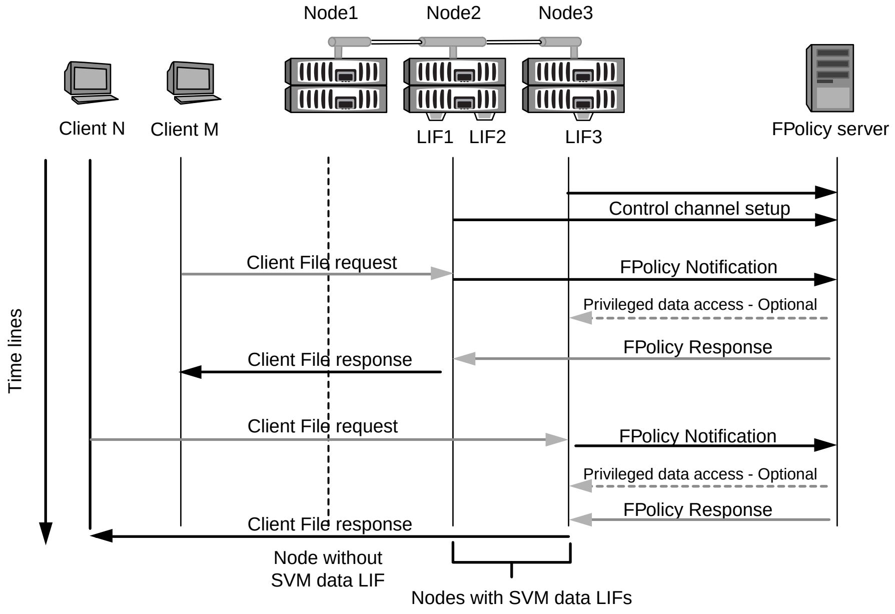
To properly plan your FPolicy configuration, you should understand what the node-to-external FPolicy server communication process is.

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, 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

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 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 extension settings configured in the FPolicy scope.

Related concepts

Planning the FPolicy policy configuration on page 70
Creating the FPolicy configuration on page 78
How FPolicy passthrough-read enhances usability for hierarchical storage management on page 52

When to create a native FPolicy configuration

Native FPolicy configurations use the 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).

How FPolicy passthrough-read enhances usability for hierarchical storage management

Passthrough-read enables the FPolicy server (functioning as the hierarchical storage management (HSM) server) to provide read access to offline files without having to recall the file from the secondary storage system to the primary storage system.

When an FPolicy server is configured to provide HSM to files residing on a CIFS server, policy-based file migration occurs where the files are stored offline on secondary storage and only a stub file remains on primary storage. Even though a stub file appears as a normal file to clients, it is actually a sparse file that is the same size of the original file. The sparse file has the CIFS offline bit set and points to the actual file that has been migrated to secondary storage.

Typically when a read request for an offline file is received, the requested content must be recalled back to primary storage and then accessed through primary storage. The need to recall data back to primary storage has several undesirable effects. Among the undesirable effects is the increased latency to client requests caused by the need to recall the content before responding to the request and the increased space consumption needed for recalled files on the primary storage.

FPolicy passthrough-read allows the HSM server (the FPolicy server) to provide read access to migrated, offline files without having to recall the file from the secondary storage system to the primary storage system. Instead of recalling the files back to primary storage, read requests can be serviced directly from secondary storage.

Note: Copy Offload (ODX) is not supported with FPolicy passthrough-read operation.

Passthrough-read enhances usability by providing the following benefits:

- Read requests can be serviced even if the primary storage does not have sufficient space to recall requested data back to primary storage.
- Better capacity and performance management when a surge of data recall might occur, such as if a script or a backup solution needs to access many offline files.
- Read requests for offline files in Snapshot copies can be serviced.
  Because Snapshot copies are read-only, the FPolicy server cannot restore the original file if the stub file is located in a Snapshot copy. Using passthrough-read eliminates this problem.
- Policies can be set up that control when read requests are serviced through access to the file on secondary storage and when the offline file should be recalled to primary storage.
  For example, a policy can be created on the HSM server that specifies the number of times the offline file can be accessed in a specified period of time before the file is migrated back to primary storage. This type of policy avoids recalling files that are rarely accessed.

Related concepts

Passthrough-read upgrade and revert considerations on page 55

How read requests are managed when FPolicy passthrough-read is enabled

You should understand how read requests are managed when FPolicy passthrough-read is enabled so that you can optimally configure connectivity between the storage virtual machine (SVM) and the FPolicy servers.

When FPolicy passthrough-read is enabled and the SVM receives a request for an offline file, FPolicy sends a notification to the FPolicy server (HSM server) through the standard connection channel.

After receiving the notification, the FPolicy server reads the data from the file path sent in the notification and sends the requested data to the SVM through the passthrough-read privileged data connection that is established between the SVM and the FPolicy server.

After the data is sent, the FPolicy server then responds to the read request as an ALLOW or DENY. Based on whether the read request is allowed or denied, ONTAP either sends the requested information or sends an error message to the client.

Requirements, considerations, and best practices for configuring FPolicy

Before you create and configure FPolicy configurations on your storage virtual machines (SVMs), you need to be aware of certain requirements, considerations, and best practices for configuring FPolicy.

Related concepts

Passthrough-read upgrade and revert considerations on page 55

Planning the FPolicy policy configuration on page 70
Creating the FPolicy configuration on page 78

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 ONTAP that supports FPolicy.
- If you are not using the 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.
    - All data LIFs used to communicate with the FPolicy servers must be configured to have **`cifs`** as one of the allowed protocols.
      This includes the LIFs used for passthrough-read connections.

Related concepts

Planning the FPolicy external engine configuration on page 57
How privileged data access channels are used for synchronous communication on page 48
How FPolicy connection credentials are used with privileged data access channels on page 48
What granting super user credentials for privileged data access means on page 49

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 that you 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.

- 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.

Related concepts

Planning the FPolicy external engine configuration on page 57

Related tasks

Enabling or disabling FPolicy policies on page 84

Passthrough-read upgrade and revert considerations

There are certain upgrade and revert considerations that you must know about before upgrading to an ONTAP release that supports passthrough-read or before reverting to a release that does not support passthrough-read.

Upgrading

After all nodes are upgraded to a version of ONTAP that supports FPolicy passthrough-read, the cluster is capable of using the passthrough-read functionality; however, passthrough-read is disabled by default on existing FPolicy configurations. To use passthrough-read on existing FPolicy configurations, you must disable the FPolicy policy and modify the configuration, and then reenable the configuration.

Reverting

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

- All the policies using passthrough-read must be disabled, and then the affected configurations must be modified so that they do not use passthrough-read.
- 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.

Related concepts

Planning the FPolicy configuration on page 56
Creating the FPolicy configuration on page 78

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). FPolicy is not supported on SVMs with Infinite Volume.

Related concepts

What the steps for setting up an FPolicy configuration are on page 55
Creating the FPolicy configuration on page 78

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.

Information that is defined when creating the FPolicy external engine

The external engine configuration defines the information that FPolicy needs to make and manage connections to the external FPolicy servers (FPolicy servers), including the following 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 using various advanced privilege settings

  This includes parameters that define such things as timeout values, retry values, keep-alive values, maximum request values, sent and receive buffer size values, and session timeout values.

The `vserver fpolicy policy external-engine create` command is used to create an FPolicy external engine.

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`*

Type of information	Option
Engine name Specifies the name to assign to the external engine configuration. You must specify the external engine name later when you create the FPolicy policy. This associates the external engine with the policy. The name can be up to 256 characters long. **Note:** The name should be up to 200 characters long if configuring the external engine name in a MetroCluster or SVM disaster recovery configuration. The name can contain any combination of the following ASCII-range characters: • **`a`** through **`z`** • **`A`** through **`Z`** • **`0`** through **`9`** • "`_`", "`-`", and "`.`"	`-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. If the external engine is used in a MetroCluster or SVM disaster recovery configuration, you should specify the IP addresses of the FPolicy servers at the source site as primary servers. The IP addresses of the FPolicy servers at the destination site should be specified as secondary servers.	`-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 using SSL server authentication. • 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 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`**}
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]
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 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]

Type of information	Option
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`*
Receive buffer size Specifies the receive buffer size of the connected socket for the FPolicy server. The default value is set to 256 kilobytes (Kb). When the value is set to 0, the size of the receive buffer is set to a value defined by the system. For example, if the default receive buffer size of the socket is 65536 bytes, by setting the tunable value to 0, the socket buffer size is set to 65536 bytes. You can use any non-default value to set the size (in bytes) of the receive buffer.	`-recv-buffer-size` *`integer`*
Send buffer size Specifies the send buffer size of the connected socket for the FPolicy server. The default value is set to 256 kilobytes (Kb). When the value is set to 0, the size of the send buffer is set to a value defined by the system. For example, if the default send buffer size of the socket is set to 65536 bytes, by setting the tunable value to 0, the socket buffer size is set to 65536 bytes. You can use any non-default value to set the size (in bytes) of the send buffer.	`-send-buffer-size` *`integer`*

Timeout for purging a session ID during reconnection
Specifies the interval in hours (**`h`**), minutes (**`m`**), or seconds (**`s`**) after which a new session ID is sent to the FPolicy server during reconnection attempts.
If the connection between the storage controller and the FPolicy server is terminated and reconnection is made within the `-session-timeout` interval, the old session ID is sent to FPolicy server so that it can send responses for old notifications.
The default value is set to 10 seconds.
	`-session-timeout`
	`[`*`integer`*`h]`
	`[`*`integer`*`m]`
	`[`*`integer`*`s]`

Related concepts

Additional information about configuring FPolicy external engines to use SSL authenticated connections on page 62

Restrictions for cluster-scoped FPolicy external engines with MetroCluster and SVM disaster recovery configurations on page 63

Related information

ONTAP concepts

Additional information about configuring FPolicy external engines to use SSL authenticated connections

You need to know some additional information if you want to configure the FPolicy external engine to use SSL when connecting to FPolicy servers.

SSL server authentication

If you choose to configure the FPolicy external engine for SSL server authentication, before creating the external engine, you must install the public certificate of the certificate authority (CA) that signed the FPolicy server certificate.

Mutual authentication

If you configure FPolicy external engines to use SSL mutual authentication when connecting storage virtual machine (SVM) data LIFs to external FPolicy servers, before creating the external engine, you 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. You must not delete this certificate while any FPolicy policies are using the installed certificate.

If the certificate is deleted while FPolicy is using it for mutual authentication when connecting to an external FPolicy server, you cannot reenable a disabled FPolicy policy that uses that certificate. The FPolicy policy cannot be reenabled in this situation even if a new certificate with the same settings is created and installed on the SVM.

If the certificate has been deleted, you need to install a new certificate, create new FPolicy external engines that use the new certificate, and associate the new external engines with the FPolicy policy that you want to reenable by modifying the FPolicy policy.

How to install certificates for SSL

The public certificate of the 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`**.

Related concepts

Planning the FPolicy external engine configuration on page 57

Certificates do not replicate in SVM disaster recovery relationships with a non-ID-preserve configuration

Security certificates used for SSL authentication when making connections to FPolicy servers do not replicate to SVM disaster recovery destinations with non-ID-preserve configurations. Although the FPolicy external-engine configuration on the SVM is replicated, security certificates are not replicated. You must manually install the security certificates on the destination.

When you set up the SVM disaster recovery relationship, the value you select for the `-identity-preserve` option of the `snapmirror create` command determines the configuration details that are replicated in the destination SVM.

If you set the `-identity-preserve` option to **`true`** (ID-preserve), all of the FPolicy configuration details are replicated, including the security certificate information. You must install the security certificates on the destination only if you set the option to **`false`** (non-ID-preserve).

Restrictions for cluster-scoped FPolicy external engines with MetroCluster and SVM disaster recovery configurations

You can create a cluster-scoped FPolicy external engine by assigning the cluster storage virtual machine (SVM) to the external engine. However, when creating a cluster-scoped external engine in a MetroCluster or SVM disaster recovery configuration, there are certain restrictions when choosing the authentication method that the SVM uses for external communication with the FPolicy server.

There are three authentication options that you can choose when creating external FPolicy servers: no authentication, SSL server authentication, and SSL mutual authentication. Although there are no restrictions when choosing the authentication option if the external FPolicy server is assigned to a data SVM, there are restrictions when creating a cluster-scoped FPolicy external engine:

Configuration	Permitted?
MetroCluster or SVM disaster recovery and a cluster-scoped FPolicy external engine with no authentication (SSL is not configured)	Yes
MetroCluster or SVM disaster recovery and a cluster-scoped FPolicy external engine with SSL server or SSL mutual authentication	No

- If a cluster-scoped FPolicy external engine with SSL authentication exists and you want to create a MetroCluster or SVM disaster recovery configuration, you must modify this external engine to use no authentication or remove the external engine before you can create the MetroCluster or SVM disaster recovery configuration.
- If the MetroCluster or SVM disaster recovery configuration already exists, ONTAP prevents you from creating a cluster-scoped FPolicy external engine with SSL authentication.

Related concepts

Planning the FPolicy external engine configuration on page 57

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		

Type of information	Required	Include	Your values
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		
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		
Receive buffer size	No		
Send buffer size	No		
Timeout for purging a session ID during reconnection	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 mount and unmount operations

Note: There is a dependency with three of the parameters (`-protocol`, `-file-operations`, `-filters`). The following combinations are valid 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 assign to the FPolicy event. When you create the FPolicy policy you associate the FPolicy event with the policy using the event name. The name can be up to 256 characters long. **Note:** The name should be up to 200 characters long if configuring the event in a MetroCluster or SVM disaster recovery configuration. The name can contain any combination of the following ASCII-range characters: • **`a`** through **`z`** • **`A`** through **`Z`** • **`0`** through **`9`** • “_”, “**`-`**”, and “**`.`**”	`-event-name` *`event_name`*

Type of information	**Option**
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`*
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: **Note:** If you specify the `-filters` parameter, then you must also specify valid values for the `-file-operations` and `-protocol` parameters. • **`monitor-ads`** option to filter the client request for alternate data stream. • **`close-with-modification`** option to filter the client request for close with modification. • **`close-without-modification`** option to filter the client request for close without modification. • **`first-read`** option to filter the client request for first read. • **`first-write`** option to filter the client request for first write. • **`offline-bit`** option 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`** option 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`** option 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`** option to filter the client request for write with size change. • **`setattr-with-owner-change`** option to filter the client setattr requests for changing owner of a file or a directory. • **`setattr-with-group-change`** option to filter the client setattr requests for changing the group of a file or a directory. • **`setattr-with-sacl-change`** option to filter the client setattr requests for changing the SACL on a file or a directory. This filter is available only for the CIFS and NFSv4 protocols. • **`setattr-with-dacl-change`** option to filter the client setattr requests for changing the DACL on a file or a directory. This filter is available only for the CIFS and NFSv4 protocols.	`-filters` *`filter`*, ...

Type of information	Option
- **`setattr-with-modify-time-change`** option to filter the client setattr requests for changing the modification time of a file or a directory. - **`setattr-with-access-time-change`** option to filter the client setattr requests for changing the access time of a file or a directory. - **`setattr-with-creation-time-change`** option to filter the client setattr requests for changing the creation time of a file or a directory. This option is available only for the CIFS protocol. - **`setattr-with-mode-change`** option to filter the client setattr requests for changing the mode bits on a file or a directory. - **`setattr-with-size-change`** option to filter the client setattr requests for changing the size of a file. - **`setattr-with-allocation-size-change`** option to filter the client setattr requests for changing the allocation size of a file. This option is available only for the CIFS protocol. - **`exclude-directory`** option to filter the client requests for directory operations. When this filter is specified, the directory operations are not monitored.	
Is volume operation required Specifies whether monitoring is required for volume mount and unmount operations. 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, close-with-read, exclude-directory
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, exclude-dir
open	monitor-ads, offline-bit, open-with-delete-intent, open-with-write-intent, exclude-dir
read	monitor-ads, offline-bit, first-read
write	monitor-ads, offline-bit, first-write, write-with-size-change
rename	monitor-ads, offline-bit

Supported file operations	Supported filters
rename_dir	Currently no filter is supported for this file operation.
setattr	monitor-ads, offline-bit, setattr_with_owner_change, setattr_with_group_change, setattr_with_mode_change, setattr_with_sacl_change, setattr_with_dacl_change, setattr_with_modify_time_change, setattr_with_access_time_change, setattr_with_creation_time_change, setattr_with_size_change, setattr_with_allocation_size_change, exclude_directory

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.
link	offline-bit
lookup	offline-bit, exclude-dir
read	offline-bit, first-read
write	offline-bit, first-write, write-with-size-change
rename	offline-bit
rename_dir	Currently no filter is supported for this file operation.
setattr	offline-bit, setattr_with_owner_change, setattr_with_group_change, setattr_with_mode_change, setattr_with_modify_time_change, setattr_with_access_time_change, setattr_with_size_change, exclude_directory
symlink	offline-bit

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, exclude-directory
create	offline-bit
create_dir	Currently no filter is supported for this file operation.

Supported file operations	Supported filters
delete	offline-bit
delete_dir	Currently no filter is supported for this file operation.
getattr	offline-bit, exclude-directory
link	offline-bit
lookup	offline-bit, exclude-directory
open	offline-bit, exclude-directory
read	offline-bit, first-read
write	offline-bit, first-write, write-with-size-change
rename	offline-bit
rename_dir	Currently no filter is supported for this file operation.
setattr	offline-bit, setattr_with_owner_change, setattr_with_group_change, setattr_with_mode_change, setattr_with_sacl_change, setattr_with_dacl_change, setattr_with_modify_time_change, setattr_with_access_time_change, setattr_with_size_change, exclude_directory
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 which parameters are required when creating the policy as well as why you might want to configure certain optional parameters. This information helps you to determine which values to set for each parameter.

When creating an FPolicy policy you associate the policy with the following:

- The storage virtual machine (SVM)
- One or more FPolicy events

- An FPolicy external engine

You can also configure several optional policy settings.

What the FPolicy policy configuration contains

You can use the following list of available FPolicy policy required and optional parameters to help you plan your configuration:

Type of information	Option	Required	Default
SVM name Specifies the name of the SVM on which you want to create an FPolicy policy.	`-vserver` *`vserver_name`*	Yes	None
Policy name Specifies the name of the FPolicy policy. The name can be up to 256 characters long. **Note:** The name should be up to 200 characters long if configuring the policy in a MetroCluster or SVM disaster recovery configuration. The name can contain any combination of the following ASCII-range characters: • **`a`** through **`z`** • **`A`** through **`Z`** • **`0`** through **`9`** • "_", "**`-`**", and "**`.`**"	`-policy-name` *`policy_name`*	Yes	None
Event names Specifies a comma-delimited list of events to associate with the FPolicy policy. • You can associate more than one event to a policy. • An event is specific to a protocol. • You can use a single 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. • The events must already exist.	`-events` *`event_name`*, ...	Yes	None

Type of information	Option	Required	Default
External engine name Specifies the name of the external engine to associate with the FPolicy policy. • An external engine contains information required by the node to send notifications to an FPolicy server. • You can configure FPolicy to use the ONTAP native external engine for simple file blocking or to use an external engine that is configured to use external FPolicy servers (FPolicy servers) for more sophisticated file blocking and file management. • 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. • If you want to use FPolicy servers, the configuration for the external engine must already exist.	`-engine` *`engine_name`*	Yes (unless the policy uses the internal ONTAP native engine)	**`native`**
Is mandatory screening required Specifies whether mandatory file access screening is required. • The mandatory screening setting determines what action is taken 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.	`-is-mandatory` {**`true`**|**`false`**}	No	**`true`**

Type of information	Option	Required	Default
Allow privileged access Specifies whether you want the FPolicy server to have privileged access to the monitored files and folders by using a privileged data connection. If configured, FPolicy servers can access files from the root of the SVM containing the monitored data using the privileged data connection. For privileged data access, CIFS must be licensed on the cluster and all the data LIFs used to connect to the FPolicy servers must be configured to have **`cifs`** as one of the allowed protocols. 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.	`-allow-privileged-access` {**`yes`**|**`no`**}	No (unless passthrough-read is enabled)	**`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`*	No (unless privileged access is enabled)	None

Type of information	Option	Required	Default
Allow passthrough-read Specifies whether the FPolicy servers can provide passthrough-read services for files that have been archived to secondary storage (offline files) by the FPolicy servers: • Passthrough-read is a way to read data for offline files without restoring the data to the primary storage. Passthrough-read reduces response latencies because there is no need to recall files back to primary storage before responding to the read request. Additionally, passthrough-read optimizes storage efficiency by eliminating the need to consume primary storage space with files that are recalled solely to satisfy read requests. • When enabled, the FPolicy servers provide the data for the file over a separate privileged data channel opened specifically for passthrough-reads. • If you want to configure passthrough-read, the policy must also be configured to allow privileged access.	`-is-passthrough-read-enabled {`**`true`**`|`**`false`**`}`	No	**`false`**

Related concepts

How FPolicy manages policy processing on page 49
Requirements, considerations, and best practices for configuring FPolicy on page 53
How FPolicy passthrough-read enhances usability for hierarchical storage management on page 52
Requirement for FPolicy scope configurations if the FPolicy policy uses the native engine on page 74

Requirement for FPolicy scope configurations if the FPolicy policy uses the native engine

If you configure the FPolicy policy to use the native engine, there is a specific requirement for how you define the FPolicy scope configured for the policy.

The FPolicy scope defines the boundaries on which the FPolicy policy applies, for example whether the FPolicy applies to specified volumes or shares. There are a number of parameters that further restrict the scope to which the FPolicy policy applies. One of these parameters, `-is-file-extension-check-on-directories-enabled`, specifies whether to check file extensions on directories. The default value is **`false`**, which means that file extensions on directories are not checked.

When an FPolicy policy that uses the native engine is enabled on a share or volume and the `-is-file-extension-check-on-directories-enabled` parameter is set to **`false`** for the scope of the policy, directory access is denied. With this configuration, because the file extensions are not checked for directories, any directory operation is denied if it falls under the scope of the policy.

To ensure that directory access succeeds when using the native engine, you must set the `-is-file-extension-check-on-directories-enabled parameter` to **`true`** when creating the scope.

With this parameter set to **`true`**, extension checks happen for directory operations and the decision whether to allow or deny access is taken based on the extensions included or excluded in the FPolicy scope configuration.

Related concepts

FPolicy configuration types on page 51
When to create a native FPolicy configuration on page 52
Planning the FPolicy scope configuration on page 75

Completing the FPolicy policy worksheet

You can use this worksheet to record the values that you need during the FPolicy policy configuration process. 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	Include	Your values
storage virtual machine (SVM) name	Yes	
Policy name	Yes	
Event names	Yes	
External engine name		
Is mandatory screening required		
Allow privileged access		
Privileged user name		
Is passthrough-read enabled		

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. If 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 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`*, ...

Type of information	Option
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`*, ...
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. If the FPolicy policy to which the scope is assigned is configured to use the native engine, this parameter must be set to **`true`**.	`-is-file-extension-check-on-directories-enabled` {**`true`**|**`false`**|}

Related concepts

Requirement for FPolicy scope configurations if the FPolicy policy uses the native engine on page 74

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	
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). FPolicy is not supported on SVMs with Infinite Volume.

Steps

1. Creating the FPolicy external engine on page 79
   You must create an external engine to start creating an FPolicy configuration. The external engine defines how FPolicy makes and manages connections to external FPolicy servers. If your configuration uses the internal ONTAP engine (the native external engine) for simple file blocking, you do not need to configure a separate FPolicy external engine and do not need to perform this step.

2. Creating the FPolicy event on page 80
   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 80
   When you create the FPolicy policy, you associate an external engine and one or more events to the policy. The policy also specifies whether mandatory screening is required, whether the FPolicy servers have privileged access to data on the storage virtual machine (SVM), and whether passthrough-read for offline files is enabled.

4. Creating the FPolicy scope on page 82
   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 82

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.

Related concepts

What the steps for setting up an FPolicy configuration are on page 55
Planning the FPolicy configuration on page 56
Requirements, considerations, and best practices for configuring FPolicy on page 53
Displaying information about FPolicy configurations on page 84
How FPolicy passthrough-read enhances usability for hierarchical storage management on page 52

Creating the FPolicy external engine

You must create an external engine to start creating an FPolicy configuration. The external engine defines how FPolicy makes and manages connections to external FPolicy servers. If your configuration uses the internal ONTAP engine (the native external engine) for simple file blocking, you do not need to configure a separate FPolicy external engine and do not need to perform this step.

Before you begin

The external engine worksheet should be completed.

About this task

If the external engine is used in a MetroCluster configuration, you should specify the IP addresses of the FPolicy servers at the source site as primary servers. The IP addresses of the FPolicy servers at the destination site should be specified as secondary servers.

Steps

1. Create the FPolicy external engine by using the `vserver fpolicy policy external-engine create` command.

   #### Example

   The following command creates an external engine on storage virtual machine (SVM) vs1.example.com. No authentication is required for external communications with the FPolicy server.

   ```
   vserver fpolicy policy external-engine create -vserver-name
   vs1.example.com -engine-name engine1 -primary-servers 10.1.1.2,10.1.1.3
   -port 6789 -ssl-option no-auth
   ```

2. Verify the FPolicy external engine configuration by using the `vserver fpolicy policy external-engine show` command.

   #### Example

   The following command display information about all external engines configured on SVM vs1.example.com:

   ```
   vserver fpolicy policy external-engine show -vserver vs1.example.com
   ```

   ```
                                   Primary        Secondary           External
   Vserver            Engine       Servers        Servers       Port Engine Type
   ---------------    -----------  -------------- ----------- ------ -----------
   vs1.example.com    engine1      10.1.1.2,      -             6789 synchronous
                                   10.1.1.3
   ```

The following command displays detailed information about the external engine named “engine1” on SVM vs1.example.com:

`vserver fpolicy policy external-engine show -vserver vs1.example.com -engine-name engine1`

```
                               Vserver: vs1.example.com
                                Engine: engine1
              Primary FPolicy Servers: 10.1.1.2, 10.1.1.3
       Port Number of FPolicy Service: 6789
            Secondary FPolicy Servers: -
                 External Engine Type: synchronous
SSL Option for External Communication: no-auth
           FQDN or Custom Common Name: -
         Serial Number of Certificate: -
                Certificate Authority: -
```

Creating the FPolicy 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 `vserver fpolicy policy event create` command.

   Example

   `vserver fpolicy policy event create -vserver-name vs1.example.com -event-name event1 -protocol cifs -file-operations open,close,read,write`

2. Verify the FPolicy event configuration by using the `vserver fpolicy policy event show` command.

   Example

   `vserver fpolicy policy event show -vserver vs1.example.com`

```
                  Event                File                     Is Volume
Vserver           Name      Protocols Operations    Filters   Operation
---------------   --------- --------- ------------- --------- ------------
vs1.example.com   event1    cifs      open, close,  -         false
                                      read, write
```

Creating the FPolicy policy

When you create the FPolicy policy, you associate an external engine and one or more events to the policy. The policy also specifies whether mandatory screening is required, whether the FPolicy servers have privileged access to data on the storage virtual machine (SVM), and whether passthrough-read for offline files is enabled.

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.
- If you want to configure privileged data access, a CIFS server must exist on the SVM.

Steps

1. Create the FPolicy policy:

   ```
   vserver fpolicy policy create -vserver-name vserver_name -policy-name
   policy_name -engine engine_name -events event_name,... [-is-mandatory
   {true|false}] [-allow-privileged-access {yes|no}] [-privileged-user-name
   domain\user_name] [-is-passthrough-read-enabled {true|false}]
   ```

   - You can add one or more events to the FPolicy policy.
   - By default, mandatory screening is enabled.
   - If you want to allow privileged access by setting the `-allow-privileged-access` parameter to **`yes`**, you must also configure a privileged user name for privileged access.
   - If you want to configure passthrough-read by setting the `-is-passthrough-read-enabled` parameter to **`true`**, you must also configure privileged data access.

   #### Example

   The following command creates a policy named “policy1” that has the event named “event1” and the external engine named “engine1” associated with it. This policy uses default values in the policy configuration:

   ```
   vserver fpolicy policy create -vserver vs1.example.com -policy-name
   policy1 -events event1 -engine engine1
   ```

   The following command creates a policy named “policy2” that has the event named “event2” and the external engine named “engine2” associated with it. This policy is configured to use privileged access using the specified user name. Passthrough-read is enabled:

   ```
   vserver fpolicy policy create -vserver vs1.example.com -policy-name
   policy2 -events event2 -engine engine2 -allow-privileged-access yes
   -privileged-user-name example\archive_acct -is-passthrough-read-enabled
   true
   ```

   The following command creates a policy named “native1” that has the event named “event3” associated with it. This policy uses the native engine and uses default values in the policy configuration:

   ```
   vserver fpolicy policy create -vserver vs1.example.com -policy-name
   native1 -events event3 -engine native
   ```

2. Verify the FPolicy policy configuration by using the `vserver fpolicy policy show` command.

   #### Example

   The following command displays information about the three configured FPolicy policies, including the following information:

   - The SVM associated with the policy
   - The external engine associated with the policy
   - The events associated with the policy
   - Whether mandatory screening is required
   - Whether privileged access is required

```
vserver fpolicy policy show
```

```
   Vserver       Policy     Events    Engine    Is Mandatory  Privileged
                 Name                                         Access
--------------   ---------  --------- --------- ------------  -----------
vs1.example.com  policy1    event1    engine1   true          no
vs1.example.com  policy2    event2    engine2   true          yes
vs1.example.com  native1    event3    native    true          no
```

Creating the FPolicy 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 using the `vserver fpolicy policy scope create` command.

   Example

   ```
   vserver fpolicy policy scope create -vserver-name vs1.example.com -
   policy-name policy1 -volumes-to-include datavol1,datavol2
   ```

2. Verify the FPolicy scope configuration by using the `vserver fpolicy policy scope show` command.

   Example

   ```
   vserver fpolicy policy scope show -vserver vs1.example.com -instance
   ```

   ```
                      Vserver: vs1.example.com
                       Policy: policy1
            Shares to Include: -
            Shares to Exclude: -
           Volumes to Include: datavol1, datavol2
           Volumes to Exclude: -
   Export Policies to Include: -
   Export Policies to Exclude: -
   File Extensions to Include: -
   File Extensions to Exclude: -
   ```

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 using the `vserver fpolicy enable` command.

   Example

   `vserver fpolicy enable -vserver-name vs1.example.com -policy-name policy1 -sequence-number 1`

2. Verify that the FPolicy policy is enabled by using the `vserver fpolicy show` command.

   Example

   `vserver fpolicy show -vserver vs1.example.com`

```
                                   Sequence
Vserver          Policy Name         Number  Status   Engine
---------------  ----------------- -------- -------- ---------
vs1.example.com  policy1                  1  on       engine1
```

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.

Related concepts

Creating the FPolicy configuration on page 78
Managing FPolicy server connections on page 87

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`
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.

Related references

Commands for displaying information about FPolicy configurations on page 85

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`***

Related tasks

Displaying information about FPolicy policy status on page 85
Displaying information about enabled FPolicy policies on page 86

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.

Related concepts

Creating the FPolicy configuration on page 78
Modifying FPolicy configurations on page 83

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 ONTAP command-line interface” section of the *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 parameters, 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`***
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
                                        Sequence
Vserver             Policy Name           Number  Status      Engine
------------------- ------------------- --------  ---------   ---------
FPolicy             cserver_policy      -         off         eng1
vs1.example.com     v1p1                -         off         eng2
vs1.example.com     v1p2                -         off         native
vs1.example.com     v1p3                -         off         native
vs1.example.com     cserver_policy      -         off         eng1
vs2.example.com     v1p1                3         on          native
vs2.example.com     v1p2                1         on          eng3
vs2.example.com     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.example.com         pol_native                native
vs1.example.com         pol_native2               native
vs1.example.com         pol1                      2
vs1.example.com         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.

Related concepts

What the two parts of the FPolicy solution are on page 46
What synchronous and asynchronous notifications are on page 46
How FPolicy works with external FPolicy servers on page 48
What the node-to-external FPolicy server communication process is on page 49

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 parameters, 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 FPolicy servers...	Enter...
That you specify	**vserver fpolicy show-engine -server *IP_address***
For a specified SVM	**vserver fpolicy show-engine -vserver *vserver_name***
That are attached with a specified policy	**vserver fpolicy show-engine -policy-name *policy_name***
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**
With the specified type	**vserver fpolicy show-engine -server-type *type*** The FPolicy server type can be one of the following: • **primary** • **secondary**
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.example.com:

```
cluster1::> vserver fpolicy show-engine -vserver vs1.example.com
FPolicy                                              Server-       Server-
Vserver         Policy    Node         Server        status        type
--------------- --------- ------------ ------------- ------------- ---------
vs1.example.com policy1   node1        10.1.1.2      connected     primary
vs1.example.com policy1   node1        10.1.1.3      disconnected  primary
vs1.example.com policy1   node2        10.1.1.2      connected     primary
vs1.example.com policy1   node2        10.1.1.3      disconnected  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.example.com policy1     10.1.1.2
node2      vs1.example.com policy1     10.1.1.2
```

Related concepts

How FPolicy works with external FPolicy servers on page 48
What the node-to-external FPolicy server communication process is on page 49

Related tasks

Displaying information about the FPolicy passthrough-read connection status on page 90

Displaying information about the FPolicy passthrough-read connection status

You can display information about FPolicy passthrough-read connection status 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 have passthrough-read data connections and for which FPolicy servers the passthrough-read connection is disconnected.

About this task

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

- SVM name
- FPolicy policy name
- Node name
- FPolicy server IP address
- FPolicy passthrough-read connection status

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 passthrough-read connection status for the cluster	**`vserver fpolicy show-passthrough-read-connection`**

If you want to display connection status information about...	Enter the command...
FPolicy passthrough-read connection status for a specified SVM	**`vserver fpolicy show-passthrough-read-connection -vserver`** ***`vserver_name`***
FPolicy passthrough-read connection status for a specified policy	**`vserver fpolicy show-passthrough-read-connection -policy-name`** ***`policy_name`***
Detailed FPolicy passthrough-read connection status for a specified policy	**`vserver fpolicy show-passthrough-read-connection -policy-name`** ***`policy_name`*** **`-instance`**
FPolicy passthrough-read connection status for the status that you specify	**`vserver fpolicy show-passthrough-read-connection -policy-name`** ***`policy_name`*** **`-server-status`** ***`status`*** The server status can be one of the following: • **`connected`** • **`disconnected`**

The following command displays information about passthrough-read connections from all FPolicy servers on the cluster:

```
cluster1::> vserver fpolicy show-passthrough-read-connection
                                            FPolicy           Server
Vserver          Policy Name   Node         Server            Status
--------------- ------------- ------------ ----------------- --------------
vs2.example.com  pol_cifs_2    FPolicy-01   2.2.2.2           disconnected
vs1.example.com  pol_cifs_1    FPolicy-01   1.1.1.1           connected
```

The following command displays detailed information about passthrough-read connections from FPolicy servers configured in the “pol_cifs_1 ” policy:

```
cluster1::> vserver fpolicy show-passthrough-read-connection -policy-name pol_cifs_1 -
instance

                                             Node: FPolicy-01
                                          Vserver: vs1.example.com
                                           Policy: pol_cifs_1
                                           Server: 1.1.1.1
                Session ID of the Control Channel: 8cef052e-2502-11e3-88d4-123478563412
                                    Server Status: connected
      Time Passthrough Read Channel was Connected: 9/24/2013 10:17:45
   Time Passthrough Read Channel was Disconnected: -
Reason for Passthrough Read Channel Disconnection: none
```

Related concepts

How FPolicy works with external FPolicy servers on page 48
How FPolicy passthrough-read enhances usability for hierarchical storage management on page 52

Related tasks

Displaying information about connections to external FPolicy servers on page 88

Using security tracing to verify or troubleshoot file and directory access

You can add permission tracing filters to instruct ONTAP to log information about why the SMB/CIFS and NFS servers on a storage virtual machine (SVM) allows or denies a client or user's request to perform an operation. This can be useful when you want to verify that your file access security scheme is appropriate or when you want to troubleshoot file access issues.

How security traces work

Security traces allow you to configure a filter that detects client operations over SMB/CIFS and NFS on the storage virtual machine (SVM), and trace all access checks matching that filter. You can then view the trace results, which provides a convenient summary of the reason that access was allowed or denied.

When you want to verify the security settings for SMB/CIFS or NFS access on files and folders on your SVM or if you are faced with an access problem, you can quickly add a filter to turn on permission tracing.

The following list outlines important facts about how security traces works:

- ONTAP applies security traces at the SVM level.
- Each incoming request is screened to see if it matches filtering criteria of any enabled security traces.
- Traces are performed for both file and folder access requests.
- Traces can filter based on the following criteria:
    - Client IP
    - SMB/CIFS or NFS path
    - Windows name
    - UNIX name
- Requests are screened for *Allowed* and *Denied* access response results.
- Each request matching filtering criteria of enabled traces is recorded in the trace results log.
- The storage administrator can configure a timeout on a filter to automatically disable it.
- If a request matches multiple filters, the results from the filter with the highest index number is recorded.
- The storage administrator can print results from the trace results log to determine why an access request was allowed or denied.

Types of access checks security traces monitor

Access checks for a file or folder are done based on multiple criteria. Security traces monitor operations on all these criteria.

The types of access checks that security traces monitor include the following:

- Volume and qtree security style
- Effective security of the file system containing the files and folders on which operations are requested
- User mapping
- Share-level permissions
- Export-level permissions
- File-level permissions
- Storage-Level Access Guard security

Considerations when creating security traces

You should keep several considerations in mind when you create security traces on storage virtual machines (SVMs). For example, you need to know on which protocols you can create a trace, which security-styles are supported, and what the maximum number of active traces is.

- You can only create security traces on SVMs.
- Each security trace filter entry is SVM specific.
  You must specify the SVM on which you want to run the trace.
- You can add permission tracing filters for SMB and NFS requests.
- You must set up the CIFS or NFS server on the SVM on which you want to create trace filters.
- You can create security traces for files and folders residing on NTFS, UNIX, and mixed security-style volumes and qtrees.
- You can add a maximum of 10 permission tracing filters per SVM.
- You must specify a filter index number when creating or modifying a filter.
  Filters are considered in order of the index number. The criteria in a filter with a higher index number is considered before the criteria with a lower index number. If the request being traced matches criteria in multiple enabled filters, only the filter with the highest index number is triggered.
- After you have created and enabled a security trace filter, you must perform some file or folder requests on a client system to generate activity that the trace filter can capture and log in the trace results log.
- You should add permission tracing filters for file access verification or troubleshooting purposes only.
  Adding permission tracing filters has a minor effect on controller performance.
  When you are done with verification or troubleshooting activity, you should disable or remove all permission tracing filters. Furthermore, the filtering criteria you select should be as specific as possible so that ONTAP does not send a large number of trace results to the log.

Performing security traces

Performing a security trace involves creating a security trace filter, verifying the filter criteria, generating access requests on an SMB or NFS client that match filter criteria, and viewing the results.

About this task

After you are finished using a security filter to capture trace information, you can modify the filter and reuse it, or disable it if you no longer need it. After viewing and analyzing the filter trace results, you can then delete them if they are no longer needed.

Steps

1. Creating security trace filters on page 94
   You can create security trace filters that detect SMB/CIFS and NFS client operations on storage virtual machines (SVMs)and trace all access checks matching the filter. You can use the results from security traces to validate your configuration or to troubleshoot access issues.
2. Displaying information about security trace filters on page 96
   You can display information about security trace filters configured on your storage virtual machine (SVM). This enables you to see which types of access events each filter traces.
3. Displaying security trace results on page 97
   You can display the security trace results generated for file operations that match security trace filters. You can use the results to validate your file access security configuration or to troubleshoot SMB and NFS file access issues.
4. Modifying security trace filters on page 98
   If you want to change the optional filter parameters used to determine which access events are traced, you can modify existing security trace filters.
5. Deleting security trace filters on page 99
   When you no longer need a security trace filter entry, you can delete it. Because you can have a maximum of 10 security trace filters per storage virtual machine (SVM), deleting unneeded filters enables you to create new filters if you have reached the maximum.
6. Deleting security trace records on page 100
   After you finish using a filter trace record to verify file access security or to troubleshoot SMB or NFS client access issues, you can delete the security trace record from the security trace log.
7. Deleting all security trace records on page 100
   If you do not want to keep any of the existing security trace records, you can delete all of the records on a node with a single command.

Creating security trace filters

You can create security trace filters that detect SMB/CIFS and NFS client operations on storage virtual machines (SVMs)and trace all access checks matching the filter. You can use the results from security traces to validate your configuration or to troubleshoot access issues.

About this task

There are two required parameters for the vserver security trace filter create command:

Required parameters	Description
`-vserver` *`vserver_name`*	*SVM name* The name of the SVM that contains the files or folders on which you want to apply the security trace filter.
`-index` *`index_number`*	*Filter index number* The index number you want to apply to the filter. You are limited to a maximum of 10 trace filters per SVM. The allowed values for this parameter are 1 through 10.

A number of optional filter parameters enable you to customize the security trace filter so that you can narrow down the results produced by the security trace:

Filter parameter	Description
`-client-ip` *`IP_Address`*	This filter specifies the IP address from which the user is accessing the SVM.
`-path` *`path`*	This filter specifies the path on which to apply the permission trace filter. The value for `-path` can use either of the following formats: • The complete path, starting from the root of the share or export • A partial path, relative to the root of the share You must use NFS style directory UNIX-style directory separators in the path value.
`-windows-name` *`win_user_name`* or `-unix-name` *`unix_user_name`*	You can specify either the Windows user name or UNIX user name whose access requests you want to trace. The user name variable is case insensitive. You cannot specify both a Windows user name and a UNIX user name in the same filter. **Note:** Even though you can trace SMB/CIFS and NFS access events, the mapped UNIX user and the mapped UNIX users' groups might be used when performing access checks on mixed or UNIX security-style data.
`-trace-allow` {**`yes`**|**`no`**}	Tracing for deny events is always enabled for a security trace filter. You can optionally trace allow events. To trace allow events, you set this parameter to **`yes`**.
`-enabled` {**`enabled`**|**`disabled`**}	You can enable or disable the security trace filter. By default, the security trace filter is enabled.
`-time-enabled` *`integer`*	You can specify a timeout for the filter, after which it is disabled.

Steps

1. Create a security trace filter:

   `vserver security trace filter create -vserver` ***`vserver_name`*** **`-index`** ***`index_number filter_parameters`***

   Example

   `filter_parameters` is a list of optional filter parameters.

For more information, see the man pages for the command.

2. Verify the security trace filter entry:

`vserver security trace filter show -vserver` ***`vserver_name`*** **`-index`** ***`index_number`***

Examples

The following command creates a security trace filter for any user accessing a file with a share path `\\server\share1\dir1\dir2\file.txt` from the IP address 10.10.10.7. The filter uses a complete path for the `-path` option. The client's IP address used to access data is 10.10.10.7. The filter times out after 30 minutes:

```
cluster1::> vserver security trace filter create -vserver vs1 -index 1 -path /dir1/dir2/
file.txt -time-enabled 30 -client-ip 10.10.10.7
cluster1::> vserver security trace filter show -index 1
Vserver  Index   Client-IP            Path            Trace-Allow  Windows-Name
-------- -----  -----------  ----------------------   -----------  -------------
vs1        1    10.10.10.7   /dir1/dir2/file.txt          no       -
```

The following command creates a security trace filter using a relative path for the `-path` option. The filter traces access for a Windows user named “joe”. Joe is accessing a file with a share path `\\server\share1\dir1\dir2\file.txt`. The filter traces allow and deny events:

```
cluster1::> vserver security trace filter create -vserver vs1 -index 2 -path /dir1/dir2/
file.txt -trace-allow yes -windows-name mydomain\joe

cluster1::> vserver security trace filter show -vserver vs1 -index 2
                                 Vserver: vs1
                            Filter Index: 2
              Client IP Address to Match: -
                                    Path: /dir1/dir2/file.txt
                       Windows User Name: mydomain\joe
                          UNIX User Name: -
                      Trace Allow Events: yes
                          Filter Enabled: enabled
               Minutes Filter is Enabled: 60
```

Displaying information about security trace filters

You can display information about security trace filters configured on your storage virtual machine (SVM). This enables you to see which types of access events each filter traces.

Step

1. Display information about security trace filter entries by using the `vserver security trace filter show` command.

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

Examples

The following command displays information about all security trace filters on SVM vs1:

```
cluster1::> vserver security trace filter show -vserver vs1
Vserver  Index   Client-IP            Path            Trace-Allow  Windows-Name
-------- -----  -----------  ----------------------   -----------  -------------
vs1        1    -            /dir1/dir2/file.txt          yes      -
vs1        2    -            /dir3/dir4/                  no       mydomain\joe
```

Displaying security trace results

You can display the security trace results generated for file operations that match security trace filters. You can use the results to validate your file access security configuration or to troubleshoot SMB and NFS file access issues.

Before you begin

An enabled security trace filter must exist and operations must have been performed from an SMB or NFS client that matches the security trace filter to generate security trace results.

About this task

You can display a summary of all security trace results, or you can customize what information is displayed in the output by specifying optional parameters. This can be helpful when the security trace results contain a large number of records.

If you do not specify any of the optional parameters, the following is displayed:

- storage virtual machine (SVM) name
- Node name
- Security trace index number
- Security style
- Path
- Reason
- User name
  The user name displayed depends on how the trace filter is configured:

If the filter is configured...	Then...
With a UNIX user name	The security trace result displays the UNIX user name.
With a Windows user name	The security trace result displays the Windows user name.
Without a user name	The security trace result displays the Windows user name.

You can customize the output by using optional parameters. Some of the optional parameters that you can use to narrow the results returned in the command output include the following:

Optional parameter	Description
`-fields` *`field_name`*, ...	Displays output on the fields you choose. You can use this parameter either alone or in combination with other optional parameters.
`-instance`	Displays detailed information about security trace events. Use this parameter with other optional parameters to display detailed information about specific filter results.
`-node` *`node_name`*	Displays information only about events on the specified node.
`-vserver` *`vserver_name`*	Displays information only about events on the specified SVM.
`-index` *`integer`*	Displays information about the events that occurred as a result of the filter corresponding to the specified index number.

Optional parameter	Description
`-client-ip` *`IP_address`*	Displays information about the events that occurred as a result of file access from the specified client IP address.
`-path` *`path`*	Displays information about the events that occurred as a result of file access to the specified path.
`-user-name` *`user_name`*	Displays information about the events that occurred as a result of file access by the specified Windows or UNIX user.
`-security-style` *`security_style`*	Displays information about the events that occurred on file systems with the specified security style.

See the man page for information about other optional parameters that you can use with the command.

Step

1. Display security trace filter results by using the `vserver security trace trace-result show` command.

   #### Example

   `vserver security trace trace-result show -user-name domain\user`

   ```
   Vserver: vs1

        Node     Index   Filter Details          Reason
        -------- ------- ---------------------   -----------------------------
        node1    3       User:domain\user        Access denied by explicit ACE
                         Security Style:mixed
                         Path:/dir1/dir2/

        node1    5       User:domain\user        Access denied by explicit ACE
                         Security Style:unix
                         Path:/dir1/
   ```

Modifying security trace filters

If you want to change the optional filter parameters used to determine which access events are traced, you can modify existing security trace filters.

About this task

You must identify which security trace filter you want to modify by specifying the storage virtual machine (SVM) name on which the filter is applied and the index number of the filter. You can modify all the optional filter parameters.

Steps

1. Modify a security trace filter:

   `vserver security trace filter modify -vserver` ***`vserver_name`*** **`-index`** ***`index_number filter_parameters`***

   - *`vserver_name`* is the name of the SVM on which you want to apply a security trace filter.
   - *`index_number`* is the index number that you want to apply to the filter. The allowed values for this parameter are 1 through 10.
   - *`filter_parameters`* is a list of optional filter parameters.

2. Verify the security trace filter entry:

vserver security trace filter show -vserver *vserver_name* -index *index_number*

Example

The following command modifies the security trace filter with the index number 1. The filter traces events for any user accessing a file with a share path `\\server\share1\dir1\dir2\file.txt` from any IP address. The filter uses a complete path for the `-path` option. The filter traces allow and deny events:

```
cluster1::> vserver security trace filter modify -vserver vs1 -index 1 -path /dir1/dir2/
file.txt -trace-allow yes

cluster1::> vserver security trace filter show -vserver vs1 -index 1
                                 Vserver: vs1
                            Filter Index: 1
              Client IP Address to Match: -
                                    Path: /dir1/dir2/file.txt
                       Windows User Name: -
                          UNIX User Name: -
                      Trace Allow Events: yes
                          Filter Enabled: enabled
               Minutes Filter is Enabled: 60
```

Deleting security trace filters

When you no longer need a security trace filter entry, you can delete it. Because you can have a maximum of 10 security trace filters per storage virtual machine (SVM), deleting unneeded filters enables you to create new filters if you have reached the maximum.

About this task

To uniquely identify the security trace filter that you want to delete, you must specify the following:

- The name of the SVM to which the trace filter is applied
- The filter index number of the trace filter

Steps

1. Identify the filter index number of the security trace filter entry you want to delete:

   vserver security trace filter show -vserver *vserver_name*

   #### Example

   vserver security trace filter show -vserver vs1

```
Vserver  Index   Client-IP            Path            Trace-Allow  Windows-Name
-------- -----  -----------  ----------------------   -----------  -------------
vs1        1    -            /dir1/dir2/file.txt          yes      -
vs1        2    -            /dir3/dir4/                  no       mydomain\joe
```

2. Using the filter index number information from the previous step, delete the filter entry:

   vserver security trace filter delete -vserver *vserver_name* -index *index_number*

   #### Example

   vserver security trace filter delete -vserver vs1 -index 1

3. Verify that the security trace filter entry is deleted:

   vserver security trace filter show -vserver *vserver_name*

Example

```
vserver security trace filter show -vserver vs1
```

```
Vserver  Index   Client-IP            Path            Trace-Allow   Windows-Name
-------- -----  -----------   ----------------------   -----------   -------------
vs1        2    -            /dir3/dir4/                 no        mydomain\joe
```

Deleting security trace records

After you finish using a filter trace record to verify file access security or to troubleshoot SMB or NFS client access issues, you can delete the security trace record from the security trace log.

About this task

Before you can delete a security trace record, you must know the record's sequence number.

Note: Each storage virtual machine (SVM) can store a maximum of 128 trace records. If the maximum is reached on the SVM, the oldest trace records are automatically deleted as new ones are added. If you do not want to manually delete trace records on this SVM, you can let ONTAP automatically delete the oldest trace results after the maximum is reached to make room for new results.

Steps

1. Identify the sequence number of the record you want to delete:

   ```
   vserver security trace trace-result show -vserver vserver_name -instance
   ```

2. Delete the security trace record:

   ```
   vserver security trace trace-result delete -node node_name -vserver
   vserver_name -seqnum integer
   ```

   Example

   ```
   vserver security trace trace-result delete -vserver vs1 -node node1 -
   seqnum 999
   ```

   - `-node` *`node_name`* is the name of the cluster node on which the permission tracing event that you want to delete occurred.
     This is a required parameter.
   - `-vserver` *`vserver_name`* is the name of the SVM on which the permission tracing event that you want to delete occurred.
     This is a required parameter.
   - `-seqnum` *`integer`* is the sequence number of the log event that you want to delete.
     This is a required parameter.

Deleting all security trace records

If you do not want to keep any of the existing security trace records, you can delete all of the records on a node with a single command.

Step

1. Delete all security trace records:

   ```
   vserver security trace trace-result delete -node node_name -vserver
   vserver_name *
   ```

- `-node` *`node_name`* is the name of the cluster node on which the permission tracing event that you want to delete occurred.
- `-vserver` *`vserver_name`* is the name of the storage virtual machine (SVM) on which the permission tracing event that you want to delete occurred.

How to interpret security trace results

Security trace results provide the reason that a request was allowed or denied. Output displays the result as a combination of the reason for allowing or denying access and the location within the access checking pathway where access is either allowed or denied. You can use the results to isolate and identify why actions are or are not allowed.

Finding information about the lists of result types and filter details

You can find the lists of result types and filter details that can be included in the security trace results in the man pages for the `vserver security trace trace-result show` command.

Example of output from the `Reason` field in an `Allow` result type

The following is an example of the output from the `Reason` field that appears in the trace results log in an `Allow` result type:

```
Access is allowed because SMB implicit permission grants requested
access while opening existing file or directory.
```

```
Access is allowed because NFS implicit permission grants requested
access while opening existing file or directory.
```

Example of output from the `Reason` field in an `Allow` result type

The following is an example of the output from the `Reason` field that appears in the trace results log in a `Deny` result type:

```
Access is denied. The requested permissions are not granted by the
ACE while checking for child-delete access on the parent.
```

Example of output from the `Filter details` field

The following is an example of the output from the `Filter details` field in the trace results log, which list the effective security style of the file system containing files and folders that match the filter criteria:

```
Security Style: MIXED and ACL
```

Where to find additional information

After you have successfully tested CIFS client access, you can perform advanced CIFS configuration or add SAN access. After you have successfully tested NFS client access, you can perform advanced NFS configuration or add SAN access. When protocol access is complete, you should protect the root volume of SVM. There are express guides, comprehensive guides, and technical reports to help you achieve these goals.

CIFS/SMB configuration

You can further configure CIFS access using the following comprehensive guides and technical reports:

- *CIFS management*
  Describes how to configure and manage file access using the CIFS/SMB protocol.
- *NetApp Technical Report 4191: Best Practices Guide for Clustered Data ONTAP 8.2 Windows File Services*
  Provides a brief overview of SMB implementation and other Windows File Services features with recommendations and basic troubleshooting information for ONTAP.
- *NetApp Technical Report 3740: SMB 2: Next-Generation CIFS Protocol in Data ONTAP*
  Describes SMB 2 features, configuration details, and its implementation in ONTAP.
- *NetApp KB Article 4550: Clustered Data ONTAP CIFS Expert Recommended articles*
  Lists all common CIFS/SMB protocol operational and troubleshooting workflows

NFS configuration

You can further configure NFS access using the following comprehensive guides and technical reports:

- *NFS management*
  Describes how to configure and manage file access using the NFS protocol.
- *NetApp Technical Report 4067: NFS Best Practice and Implementation Guide*
  Serves as an NFSv3 and NFSv4 operational guide and provides an overview of ONTAP operating system with a focus on NFSv4.
- *NetApp Technical Report 4379: Name Services Best Practices Guide*
  Provides a comprehensive list of best practices, limits, recommendations, and considerations when configuring LDAP, NIS, DNS, and local user and group files for authentication purposes.
- *NetApp Technical Report 4073: Secure Unified Authentication*
  Explains how to configure ONTAP for use with UNIX-based Kerberos version 5 (krb5) servers for NFS storage authentication and Windows Server Active Directory (AD) as the KDC and Lightweight Directory Access Protocol (LDAP) identity provider.
- *NetApp Technical Report 3580: NFSv4 Enhancements and Best Practices Guide: Data ONTAP Implementation*
  Describes the best practices that should be followed while implementing NFSv4 components on AIX, Linux, or Solaris clients attached to systems running ONTAP.

Root volume protection

After configuring protocols on the SVM, you should ensure that its root volume is protected:

- *Data protection*
  Describes how to create a load-sharing mirror to protect the SVM root volume, which is a NetApp best practice for NAS-enabled SVMs. Also describes how to quickly recover from volume failures or losses by promoting the SVM root volume from a load-sharing mirror

Copyright

Trademark

NETAPP, the NETAPP logo, and the marks listed on the NetApp Trademarks page are trademarks of NetApp, Inc. Other company and product names may be trademarks of their respective owners.

http://www.netapp.com/us/legal/netapptmlist.aspx

How to send comments about documentation and receive update notifications

You can help us to improve the quality of our documentation by sending us your feedback. You can receive automatic notification when production-level (GA/FCS) documentation is initially released or important changes are made to existing production-level documents.

If you have suggestions for improving this document, send us your comments by email.

doccomments@netapp.com

To help us direct your comments to the correct division, include in the subject line the product name, version, and operating system.

If you want to be notified automatically when production-level documentation is released or important changes are made to existing production-level documents, follow Twitter account @NetAppDoc.

You can also contact us in the following ways:

- NetApp, Inc., 1395 Crossman Ave., Sunnyvale, CA 94089 U.S.
- Telephone: +1 (408) 822-6000
- Fax: +1 (408) 822-4501
- Support telephone: +1 (888) 463-8277

Index