Technical Report

Clustered Data ONTAP CIFS Auditing Quick Start Guide

Sharyathi Nagesh, NetApp
February 2015 | TR-4189

Summary

This technical report discusses the native auditing implementation in the NetApp® clustered Data ONTAP® operating system with specific focus on the Common Internet File System (CIFS). This document serves as a reference for customers and partners who want to use this feature. Native auditing helps to monitor file activities in NAS environments for diagnostic or reporting purposes. This report covers information on audit configuration, event support, and log format.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

1 Introduction

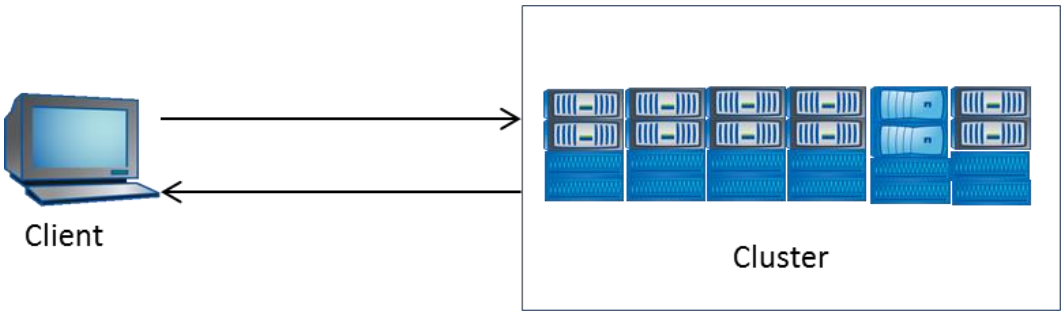
Native auditing helps to generate and manage file access logs on NetApp controllers. This feature helps to meet industry requirements such as compliance, secure log management, and intrusion detection. A storage administrator can use this feature to monitor CIFS/NFS user activities on files and folders.

Native auditing implementation for clustered Data ONTAP is supported from version 8.2 onward. This report describes how to configure auditing in clustered Data ONTAP, access log files, and interpret log information. Native auditing provides a file auditing framework that supports both CIFS and NFS protocols. Auditing in CIFS is based on NTFS, system access control lists (SACLs), or NFS 4.x access control lists (ACLs). This document focuses exclusively on auditing in CIFS file activity and best practices. For NFS-specific auditing information, refer to TR-4067: Clustered Data ONTAP NFS Best Practice and Implementation Guide.

1.1 Introduction to Clustered Data ONTAP

Clustered Data ONTAP supports scale-out architecture that can be used to add multiple NetApp nodes that provide scalability for storage capacity and performance.

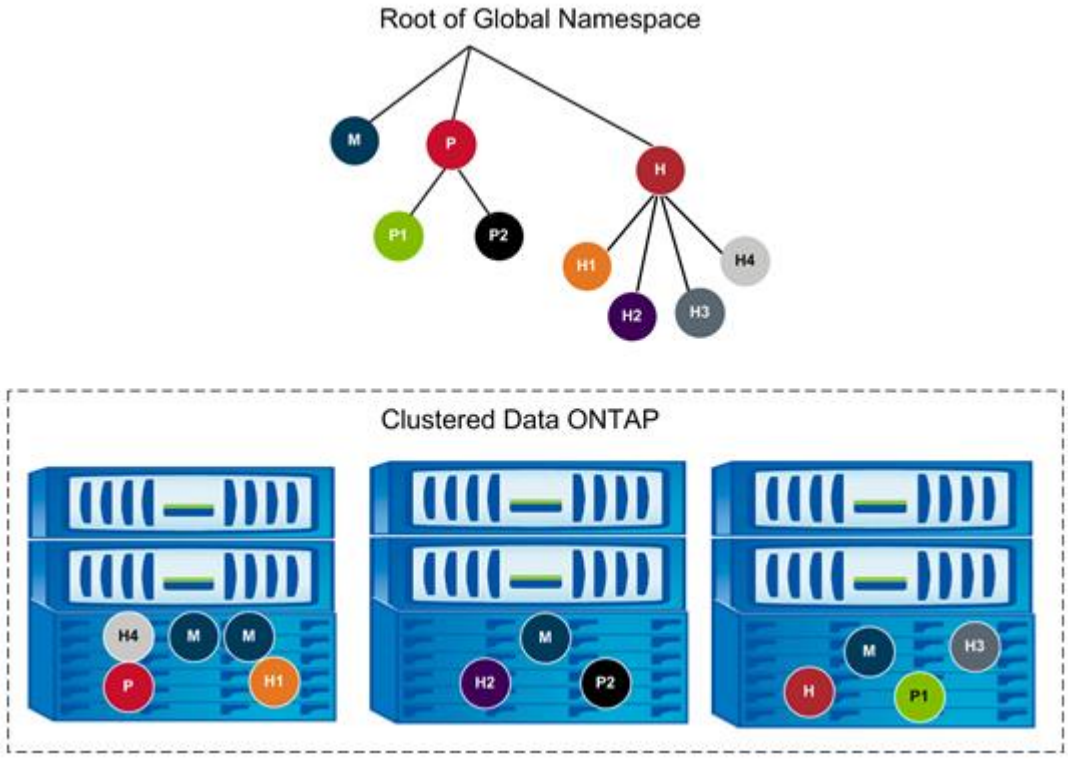
Figure 1) Data ONTAP: a scale-out architecture.



1.2 Introduction to Data ONTAP Global Namespace

The namespace offered by a storage virtual machine (SVM, formerly called Vserver) is called a NetApp global namespace. It acts as a container for all storage object servers by the SVM and identifies each such object with a unique identity. NetApp global namespace supports combining volumes across the cluster to provide a single namespace. Junction points provide means to join volumes together, creating a single namespace. This capability provides additional flexibility in laying out namespaces when compared to Data ONTAP 7-Mode.

Figure 2) Global namespace in clustered Data ONTAP.



The global namespace created using junction points has the following characteristics:

- Stitching volumes together is transparent to the clients.
- CIFS shares can be created on volumes, qtrees, or folders.

1.3 Introduction to Data ONTAP Native Auditing Implementation

The native auditing framework enables a storage administrator to monitor user actions such as access and modification of data files. This framework can be quickly configured to monitor file activities for both compliance needs as well as for short-term diagnostic purposes.

The native auditing feature in Data ONTAP 8.2 supports both CIFS and NFS protocols. A CIFS or an NFS license is required to configure this feature. To support reliable auditing, the audit information is stored on the disk instead of in memory so that in the event of a node or cluster crash, the latest audit information is committed to the disk.

To enhance performance and the user experience, this audit information is stored in a specific location in each aggregate. This location is referred to as the staging volume. The log records in the staging volume are consolidated into a single log file on a periodic basis. The location and consolidated log file are specified during audit configuration. This process is explained in section 2.1.

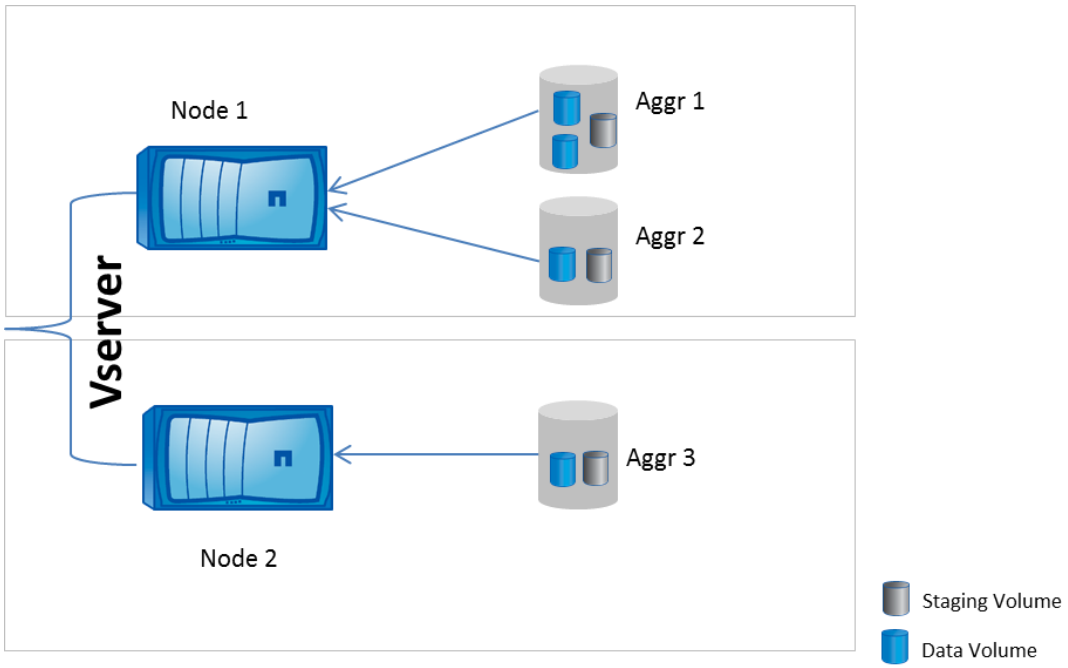
The creation of staging volumes is transparent to end users. After the creation of an audit policy on any one SVM in the cluster, a staging volume is created on all the aggregates in the cluster. From then on, all other SVMs use the existing staging volumes. Each staging volume consumes 2GB of free space and needs to be provisioned during configuration.

Staging volumes are created under a cluster SVM context, not under the data SVM context. A staging volume can be accessed only by the Cserver administrator. The Cserver administrator can resize the staging volume in diag-mode by using the `vol resize` option.

If the staging volume gets filled up, the CIFS operations will be blocked. NetApp recommends following the best practices for configuring log rotation, destination volumes, and guaranteed auditing that are listed in TR-4191: Best Practices Guide for Clustered Data ONTAP 8.2.x and 8.3 Windows File Services. NetApp does not recommend resizing or changing the staging volume because the size is determined after extensive deliberation.

For example, in Figure 3, the SVM is spread across two nodes and three aggregates. Enabling auditing would create a staging volume in each of the aggregates and by default would take up 2GB of space for each aggregate.

Figure 3) Staging volume creation in clustered Data ONTAP.



Log consolidation is scheduled every 10 seconds, and scheduling depends on the available CPU bandwidth in the user space. Log consolidation cannot be configured. This point is explained in section 2.1.

2 Configuration of Native Auditing

This section introduces the configuration required to enable auditing on clustered Data ONTAP for SVM context and configuration of SACLs on files and folders.

2.1 Configuration of Native Auditing on Data ONTAP CLI

The `SVM audit` command enables or disables auditing, defines log location files, manages log rotation, and so on.

You can configure auditing by using either the cluster admin or SVM vsadmin credential. With the cluster admin credential, you can apply configuration to any SVM in the cluster; the SVM vsadmin credential restricts you to only the specific SVM context. The examples in this report are for the cluster context. In the cluster context, you can access/modify/create the audit config for all the SVMs in the cluster. In the SVM context, you can access only the SVM audit config.

Additionally, enabling auditing requires the configuration of SACLs. To configure SACLs on files and folders, users needs to have SeSecurity privileges. By default, only the local user BUILTIN\administrator has this privilege.

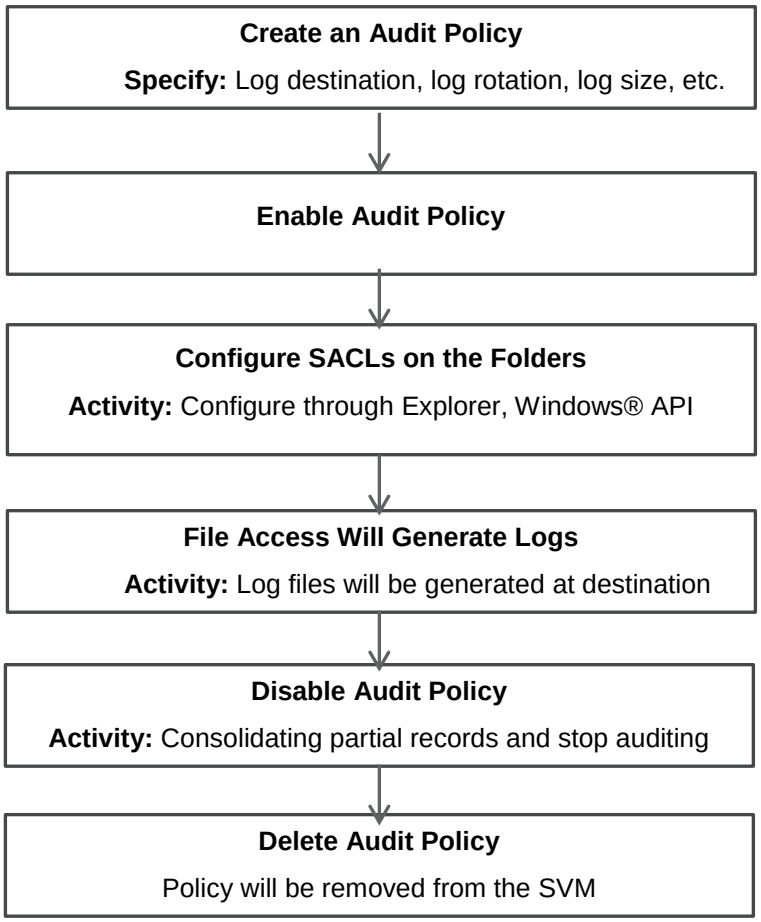
To assign SeSecurity privileges to a user, run the following command:

```
vserver cifs users-and-groups privilege add-privilege -vserver <vserver name> -user-or-group-name
<user> -privileges SeSecurityPrivilege
```

Workflow for Configuring Audit Policy on SVM

The following flow chart captures the workflow for enabling native auditing in clustered Data ONTAP 8.2 and later. This report primarily explains audit configuration through the CLI; equivalent operations are possible through the NetApp ONTAPI® library as well. To configure through ONTAPI, refer to the Appendix.

Figure 4) Configure audit policy workflow.



Create an Audit Policy on SVM

The first step for enabling auditing on an SVM is to create an audit policy. The SVM name, destination path for saving logs, and log rotation parameters are required as inputs. You can create only one active policy for each SVM. This command will either:

- Create new staging volumes if the staging volume does not already exist in the data aggregate.
- Share an existing staging volume in the data aggregate without compromising on multi-tenancy. In some instances, the staging volume can be shared by multiple SVMs.

By default, the staging volume consumes 2GB of space. The audit will fail if there's insufficient free space on the aggregate in which the data volume resides.

Creating Policy Based on Log Size

In the following example, an audit policy is created for the specified SVM with log location specified in the destination field. The destination path is a path to the folder location and should have been created previously. The size of the log file is specified through the rotate-size field. The rotate-limit parameter specifies the maximum number of log files that will be retained in the specified destination. Log files beyond this value will be overwritten. A value of zero indicates unlimited log files; in this case, the number of log files will be limited by the available free space in the destination. NetApp does not recommend setting this value to zero. When the destination volume is filled up, the CIFS client operations will be affected.

```
vserver audit create -vserver <vserver_name> -destination <unix path> -rotate-size <size MB> -
rotate-limit <Number of log files>
```

When log files reach the specified rotate size, the action triggers log rotation. The log rotate size should be greater than 1024KB and the default value is 100MB.

Creating Policy Based on Time

This example illustrates creating an audit policy for a specified SVM with the log location specified in the destination field by using the following command:

```
vserver audit create -vserver <vserver_name> -destination <unix path> -rotate-schedule-minute
<minute of the hour> -rotate-limit <Number of log files>
```

Note: The `rotate-limit` parameter specifies the maximum number of log files that will be kept in the specified destination.

The `rotate-schedule` parameter defines how often the audit log file will be rotated. For more details about log rotation, refer to section 3.3.

Enable an Audit Policy on SVM

After the audit policy is created, it needs to be enabled for audit action to begin. To enable the audit policy, use the following command:

```
vserver audit enable -vserver <vserver_name>
```

Disable an Audit Policy on SVM

This command consolidates any partial audit records present in the staging volumes into the consolidated audit log file and stops further logging of audit records:

```
vserver audit disable -vserver <vserver_name>
```

Delete an Audit Policy on SVM

Deleting the audit policy can free space for data by deleting the staging volume. If the staging volume is used by another SVM, deletion is not possible. Staging volumes are deleted only when all the SVM references are deleted.

An audit policy can be deleted by using the following command:

```
vserver audit delete -vserver <vserver_name>
```

Modify an Audit Policy on SVM

The `audit modify` command modifies the parameters previously created for an audit policy. This command can be used to modify log destination location and rotation policies such as number of concurrent log files, log rotation triggers, and so on.

```
vserver audit modify -vserver <vserver_name> -destination <unix path> -rotate-size 100MB -rotate-limit 0
```

2.2 Configuration of SACLs on the Storage Object

After enabling the audit policy at the SVM level, configure SACLs on files, folders, or shares.

SACLs can be configured on files and folders as follows:

- By using client applications such as Windows Explorer
- From script/application using appropriate Windows APIs
- From file-directory (Fsecurity) command through the CLI

SACLs can be configured on shares as follows:

- By setting SACLs on the root of the share from the Windows client

Note: Windows RPCs are currently not supported. Configuration through MMC or a dependent application is not possible.

2.3 Supported Audit Events

The auditing framework supports the logging of file and folder access operations. Table 1 lists the equivalent Windows object access operation ID. Both success auditing and failure auditing are supported for each of these operations.

Table 1 lists the supported events. The mapping of these events and the Windows events is on a best-effort basis. Some of the information present in a Windows event might not be provided in the Data ONTAP environment; for example, Windows audit records capture process ID and process name, which is not possible in Data ONTAP audit records.

Access Events Supported in Data ONTAP 8.2

Native auditing was introduced in the first release of clustered Data ONTAP 8.2. We provided support to basic audit events that will help in tracking file operations and generating required audit trails.

Table 1) Supported access events in Data ONTAP 8.2.

Windows Event ID	Event Name	Description
4656	Open object	A handle to an object is requested. This corresponds to event ID 560 in Windows Server® 2003 (W2k3) and earlier.
	Create object	
4663	Read object	An attempt was made to access an object. This corresponds to event ID 567 in W2k3 and before. This event documents the operations performed against data objects. This event logs operations that take place between the open and the close events for the object. Read and write events are optimized to log only the first read and write to make them more effective.
	Write object	
	Get object attributes	
	Set object attributes	
4664	Hard link	An attempt was made to create a hard link. A hard link is a pointer to another file in the same file system.

Windows Event ID	Event Name	Description
9999	Rename object	Added by NetApp. This ID captures the object rename operation. This is currently not supported by Windows as a single event.
9998	Unlink object	Added by NetApp. This ID captures the object unlink operation. This is currently not supported by Windows as a single event.

Note: NetApp does not support the close object event, event ID 4658, because it was creating unwanted notifications.

Note: In Data ONTAP 8.2, the monitoring delete operation is supported only through event ID 4656. The event has all the information required for identifying the delete event. The event has desired access fields that specify if the file is opened with delete intent, helping to identify delete operations.

Access Events Supported in Data ONTAP 8.2 P2

The SMB protocol supports two methods of deleting files. This support was provided by adding the two additional events listed in Table 2. NetApp strongly recommends deploying Data ONTAP 8.2 P2 and higher to leverage the benefits of these additional events.

Table 2) Supported access events in Data ONTAP 8.2 P2.

Windows Event ID	Event Name	Description
4659	Object delete	A handle to object is requested with intent to delete. It corresponds to event 563 in W2K3.
4660	Object delete	This event is generated when the object under consideration is deleted. It corresponds to event 564 in W2K3.

Access Events Supported in Data ONTAP 8.3

Two additional categories of events introduced in clustered Data ONTAP 8.3 are:

- CIFS logon/logoff events
- Central Access Policy (CAP) staging events

Table 3) Supported logon/logoff events in Data ONTAP 8.3.

Windows Event ID	Event Name	Description
4624	Local user/Network user logon	An account was successfully logged on and a CIFS session is established. It corresponds to event 528 and 540 in W2K3.
4625	Logon failures	An account was unsuccessful in logging and establishing a CIFS session. It corresponds to event 529–537 and 539 in W2K3.
4634	Local user/Network user logoff	An account was successfully logged out and a CIFS session is disconnected. It corresponds to event 538 in W2K3.

Table 4) Supported Central Access Policy events in Data ONTAP 8.3.

Windows Event ID	Event Name	Description
4818	Object access, central policy staging	These sets of events are used to evaluate the impact of Central Access Policies configured through AD and applied through the group policy objects on SVMs.

Auditing of these events can be enabled during audit policy configuration starting from Data ONTAP 8.3 onward.

```
vserver audit create -vserver <vserver_name> -destination <unix path> -events cap-staging,file-ops,cifs-logon-logoff -format evtx -rotate-size <size MB> -rotate-limit <Number of log files>
```

For more information about describing security events, refer to MS KB Article ID: 947226: Description of security events in Windows Vista and in Windows Server 2008.

3 Managing Audit Logs

3.1 Audit Log File Format

In Data ONTAP 8.2, audit logs are generated in XML format only. To convert the audit log to Windows EVTX format, use the off-box tool **NetApp EVTX Converter.** You can find more information about the EVTX Converter in the community blog.

Starting with clustered Data ONTAP 8.2.1, both XML and EVTX formats are supported as log formats. They are provided as options during audit policy configuration. EVTX is used as the default log format unless a different format is specified during audit policy configuration.

You can change the audit log file format to the EVTX format during audit policy configuration starting with Data ONTAP 8.2.1.

```
vserver audit create -vserver <vserver_name> -destination <unix path> -format evtx -rotate-size <size MB> -rotate-limit <Number of log files>
```

Log File Naming Convention

The following is the naming convention of the consolidated log file format, which cannot be configured:

```
audit_<vservername>_D<yyyy>-<MM>-<DD>-T<HH>-<MM>-<SS>_milliseconds.Xml
```

3.2 Audit Log Record Format

The file audit records are saved in an audit log file. The records follow a format similar to the Windows event framework.

Schema of Log Records

Although the log record format is closely aligned with the Windows EVTX format, it follows the NetApp proprietary format. This was done to accommodate the unique nature of the underlying Data ONTAP framework and to improvise on the existing framework wherever possible.

Detailed documentation of the event schema is shared in the community link.

Path of File in Notifications

The path information provided in logs will include only the relative path from the root of the containing volume. The user needs to construct the absolute path information from the volume ID, also called msID, and the information available in the file handler field of the log record.

Here is an example:

If there are two volumes—vol0 and vol1—with vol0 joined on / and vol1 on /home/userA, the path /home/userA/division/team/prod has /home/userA in vol0 and /division/team/prod in vol1.

When the file in /home/userA/division/team/prod is accessed, only the path /division/team/prod is available in the notification. The mount point of the volume vol1, which is /home/userA, is called the junction point of the volume vol1.

To construct the absolute path name, the information available outside the log records must be used. Clustered Data ONTAP can be queried with a `volume-get-iter` ONTAPI call with unique msID to retrieve its junction point. A user developing this support can cache the msID to junction path mapping to avoid calling it every time. Since the namespace will not change frequently, one-time operation to build the namespace should be sufficient.

Note: When a new volume is added, the SVM has to be queried again to find the junction point. In rare instances, if the volumes are remounted on a new junction path, the global namespace will be changed. In such instances, periodic querying with `volume-get-iter` to update the volume–junction path mapping is required.

3.3 Audit Log Rotation

The audit log rotation feature rotates the active log files to which the audit records are written. The log rotation can be configured for time or size.

If the log size and log rotation parameters are not specified, the default values will be used. The default value is log rotation based on a log size of 100MB. New logs will be created until the destination volume has free space. The number of concurrent files kept for log management can be changed with the rotate-limit parameter.

Log Rotation Based on Time

Log rotation is based on calendar date and time. The parameters supported are:

- Month
- Day
- Time: Specific hour and minute of the day. Specifying in minutes is mandatory. For example, on specifying the minute field as 45, at every 45th minute of the hour a new log file will be generated.

The following command creates new log files on specific days of the week:

```
vserver audit modify -vserver <vserver_name> -destination <unix path> -rotate-schedule-month
February,March -rotate-schedule-dayofweek Sunday -rotate-schedule-hour 22 -rotate-schedule-minute
45 -rotate-limit <Number of log files>
```

Log rotation can be based on calendar date and time. The parameters supported are:

```
vserver audit modify -vserver <vserver_name> -destination <unix path> -rotate-schedule-month
February,March -rotate-schedule-day 22 -rotate-schedule-hour 22 -rotate-schedule-minute 45 -
rotate-limit <Number of log files>
```

Log Rotation Based on Log Size

Log rotation can be based on log size. This can be configured using the following command:

```
vserver audit modify -vserver <vserver_name> -destination <unix path> -rotate-size <size MB> -
rotate-limit <Number of log files>
```

3.4 Accessing Audit Logs

Audit logs will be saved in the destination location specified during audit configuration. The logs can be accessed over the data access path. The destination path and the file can be accessed through CIFS shares. Access can be restricted with share-level ACLs or through folder- or file-level ACLs. Similar access is possible through the NFS export path as well.

Note: Access to audit logs is through a pull mechanism and retrieved over NFS, CIFS, or another file access protocol method. Audit logs are not integrated with the syslog framework and hence logs cannot be accessed through the push mechanism.

3.5 Partial Logs

During cluster failovers, the audit engine cannot consolidate the complete Vserverized logs. In this case, the audit log file name will indicate that it is a partial file. As soon as the node boots up, the audit engine will consolidate the records and order them chronologically.

Appendix

Audit Guarantee Feature

This feature supports guaranteed logging of audit events. This action is useful when auditing is highly critical, either because of organizational policies or because of regulatory requirements. The feature enables log records to be written to disk before file operations are completed, leaving a highly reliable audit trail. Enabling guaranteed auditing without following the auditing best practices can cause client disruptions. In case records cannot be committed to the disk because of insufficient space in the staging volume or the destination volume client, I/Os will be blocked. This feature is enabled by default and therefore care should be taken when configuring log rotation and destination volume size. They need to be configured as per the best practices listed in TR-4191: Best Practices Guide for Clustered Data ONTAP 8.2.x and 8.3 Windows File Services.

This feature can be configured in diag-mode as follows:

```
vserver audit modify -vserver <vserver_name> -destination <unix path> -rotate-size 100MB -rotate-
limit 0 -audit-guarantee true|false
```

Performance Impact of Auditing

Enabling auditing on CIFS has marginal impact on latency and CPU utilization. NetApp completed extensive testing to characterize the performance impact of auditing and recommends following our best practices to minimize the performance impact.

Enabling auditing on multiple SVMs within a single cluster will affect performance. We tested Data ONTAP 8.2.1 with 50 SVMs with minimal performance impact. Consider the number of audit-enabled SVMs before deploying the auditing feature.

Relevant ONTAPI Interfaces for Configuring Auditing

The auditing features can be configured either through the command-line interface (CLI) or through APIs. Data ONTAP APIs (ONTAPI or the NetApp Manage ONTAP® storage development kit) supported with auditing allow configuring auditing remotely. Information about ONTAPI interfaces can be found in NM-SDK documentation available at the NetApp Developer Community. The developer forum is a useful reference for developers with technical queries.

Note: Cluster ONTAPI interfaces are supported from NM-SDK 4.2 and later.

Table 4) List of audit ONTAPI interfaces added to clustered Data ONTAP.

ONTAPI Interfaces	Description
fileservice-audit-config-get	Provides audit configuration details for a particular SVM.
fileservice-audit-config-get-total-records	Obtains the total number of audit configuration entries/records in the table.
fileservice-audit-config-get-iter	Provides audit configuration details for all the SVMs.
fileservice-audit-config-create	Creates audit configuration for a particular SVM.
fileservice-audit-config-destroy	Deletes audit configuration for a particular SVM.
fileservice-audit-config-modify	Modifies the audit configuration for a particular SVM.
fileservice-audit-enable	Enables auditing for a particular SVM.
fileservice-audit-disable	Disables auditing for a particular SVM.

Executing the ONTAPI Interfaces

Although some of the ONTAPI interfaces run only in clustered context and some only in SVM context, a few ONTAPI interfaces run in both. Keep this in mind before calling ONTAPI.

- **Cluster APIs.** These APIs are executed against the `cluster-mgmt` IP using cluster administration credentials.
- **Vserver APIs.** These APIs are executed using one of the following options:
  - Calling ONTAPI against an SVM LIF using SVM admin credentials
  - Calling ONTAPI against the cluster-mgmt IP with cluster admin credentials, but using tunneling

Using Fsecurity to Set SACLs on Files and Folders

SACLs can be configured from the `vserver security file-directory` command family through the CLI. The command family was known as the Fsecurity feature in Data ONTAP 7-Mode.

This command can be used to construct Security Descriptor Definition Language and applied across multiple folders and files. Applying SACLs using this command is quicker than when applying it over the Windows host.

For more information about the command usage, refer to File Access Management Guide for CIFS.

References

The following references were used in this technical report:

- File Access Management Guide for CIFS
- Description of security events in Windows Vista® and Windows Server 2008: http://support.microsoft.com/kb/947226
- TR-4191: Best Practices Guide for Clustered Data ONTAP 8.2.x and 8.3 Windows File Services
- TR-4067: Clustered Data ONTAP NFS Best Practice and Implementation Guide

Refer to the Interoperability Matrix Tool (IMT) on the NetApp Support site to validate that the exact product and feature versions described in this document are supported for your specific environment. The NetApp IMT defines the product components and versions that can be used to construct configurations that are supported by NetApp. Specific results depend on each customer's installation in accordance with published specifications.