



ONTAP® 9

Archive and Compliance Using SnapLock® Technology Power Guide

November 2017 | 215-11235_F0
doccomments@netapp.com

Updated for ONTAP 9.3

 **NetApp®**

Contents

Deciding whether to use this Guide	4
What SnapLock is	5
SnapLock workflow	7
Configuring SnapLock	8
Installing the license	8
Initializing the ComplianceClock	8
Creating a SnapLock aggregate	9
Creating a SnapLock volume	10
Mounting a SnapLock volume	11
Setting the retention time	11
Setting the retention time for a file explicitly	12
Setting the default retention period	12
Verifying SnapLock settings	14
Committing files to WORM	16
Committing files to WORM manually	16
Autocommitting files to WORM	16
Creating a WORM appendable file	17
Using Volume Append Mode to create WORM appendable files	18
Committing Snapshot copies to WORM	20
Mirroring WORM files	24
Using the privileged delete feature	27
Creating an audit log	27
Creating a compliance administrator account	28
Enabling the privileged delete feature	29
Deleting WORM files using privileged delete	29
Holding tamper-proof files for an indefinite period	30
Using the Event Based Retention (EBR) feature	31
SnapLock APIs	32
Where to find additional information	34
Copyright information	35
Trademark information	36
How to send comments about documentation and receive update notifications	37
Index	38

Deciding whether to use the Archive and Compliance Power Guide

This guide describes how to use NetApp SnapLock technology to retain files in unmodified form for regulatory and governance purposes. It shows you how to commit files and Snapshot copies to “write once, read many” (WORM) storage, and how to set retention periods for WORM-protected data.

You should use this guide if you want to work with SnapLock in the following ways:

- You want to use the ONTAP command-line interface (CLI), not OnCommand System Manager or an automated scripting tool.
A limited but important set of SnapLock technology is available in System Manager. You can install SnapLock licenses, set the Compliance Clock, create SnapLock aggregates and volumes, and configure SnapLock volumes.
- You are not using Infinite Volumes.
Infinite Volumes are not supported by SnapLock.
- You are not using ONTAP Select.
ONTAP Select does not support SnapLock.
- You are not creating Compliance aggregates for MetroCluster configurations or array LUNs.
SnapLock Compliance aggregates do not support MetroCluster configurations or array LUNs.
- You are not creating Compliance aggregates with the SyncMirror option.
SnapLock Compliance aggregates do not support SyncMirror plexes.
- You are not planning to use the privileged delete feature for files on Enterprise volumes in MetroCluster configurations.
SnapLock Enterprise aggregates do not support the privileged delete feature in MetroCluster configurations.

Note: SSDs and Flash Pool aggregates are supported by SnapLock starting with ONTAP 9.1.

For information on how to revert from ONTAP 9.1 to ONTAP 9.0 after you have created a SnapLock SSD or Flash Pool aggregate, see the *Upgrade and Revert/Downgrade Guide*.

[Upgrade, revert, or downgrade](#)

If this guide is not suitable for your situation, you should see the following documentation instead:

- [ONTAP 9 commands](#)
- [Cluster management using System Manager](#)
- [NetApp Documentation: OnCommand Workflow Automation \(current releases\)](#)

What SnapLock is

SnapLock is a high-performance compliance solution for organizations that use WORM storage to retain files in unmodified form for regulatory and governance purposes. A single license entitles you to use SnapLock in strict *Compliance mode*, to satisfy external mandates like SEC Rule 17a-4, and a looser *Enterprise mode*, to meet internally mandated regulations for the protection of digital assets.

Differences between Compliance and Enterprise modes

SnapLock Compliance and Enterprise modes differ mainly in the level at which each mode protects WORM files:

- Compliance-mode WORM files are protected at the disk level.
You cannot reinitialize a disk that contains Compliance-mode aggregates.
- Enterprise-mode WORM files are protected at the file level.

A related difference involves how strictly each mode manages file deletes:

- Compliance-mode WORM files cannot be deleted during the retention period.
- Enterprise-mode WORM files can be deleted during the retention period by the compliance administrator, using an audited *privileged delete* procedure.

After the retention period has elapsed, you are responsible for deleting any files you no longer need. Once a file has been committed to WORM, whether under Compliance or Enterprise mode, it cannot be modified, even after the retention period has expired.

The following table shows the differences between SnapLock Compliance and Enterprise modes:

Capability	SnapLock Compliance	SnapLock Enterprise
Privileged delete	No	Yes
Reinitialize disk	No	Yes
Destroy SnapLock aggregate and volume during retention period	No	Yes
Rename an aggregate or volume	No	Yes
Non-NetApp disks support	No	Yes (with FlexArray Virtualization)
Use SnapLock volume for audit logging	Yes	No
Single-file SnapRestore	No	Yes
SnapRestore	No	Yes
MetroCluster support	Yes	Yes (except for privileged delete)

Committing files to WORM

You can use an application to commit files to WORM over NFS or CIFS, or use the SnapLock autocommit feature to commit files to WORM automatically. You can use a *WORM appendable file* to retain data that is written incrementally, like log information.

Data protection

SnapLock supports data protection methods that should satisfy most compliance requirements:

- You can use SnapLock for SnapVault to WORM-protect Snapshot copies on secondary storage.
- You can use SnapMirror to replicate WORM files to another geographic location for disaster recovery.

7-Mode Transition

You can use the Copy-Based Transition (CBT) feature of the 7-Mode Transition Tool to migrate SnapLock volumes from 7-Mode to ONTAP. The SnapLock mode of the destination volume, Compliance or Enterprise, must match the SnapLock mode of the source volume. You cannot use Copy-Free Transition (CFT) to migrate SnapLock volumes.

Encryption

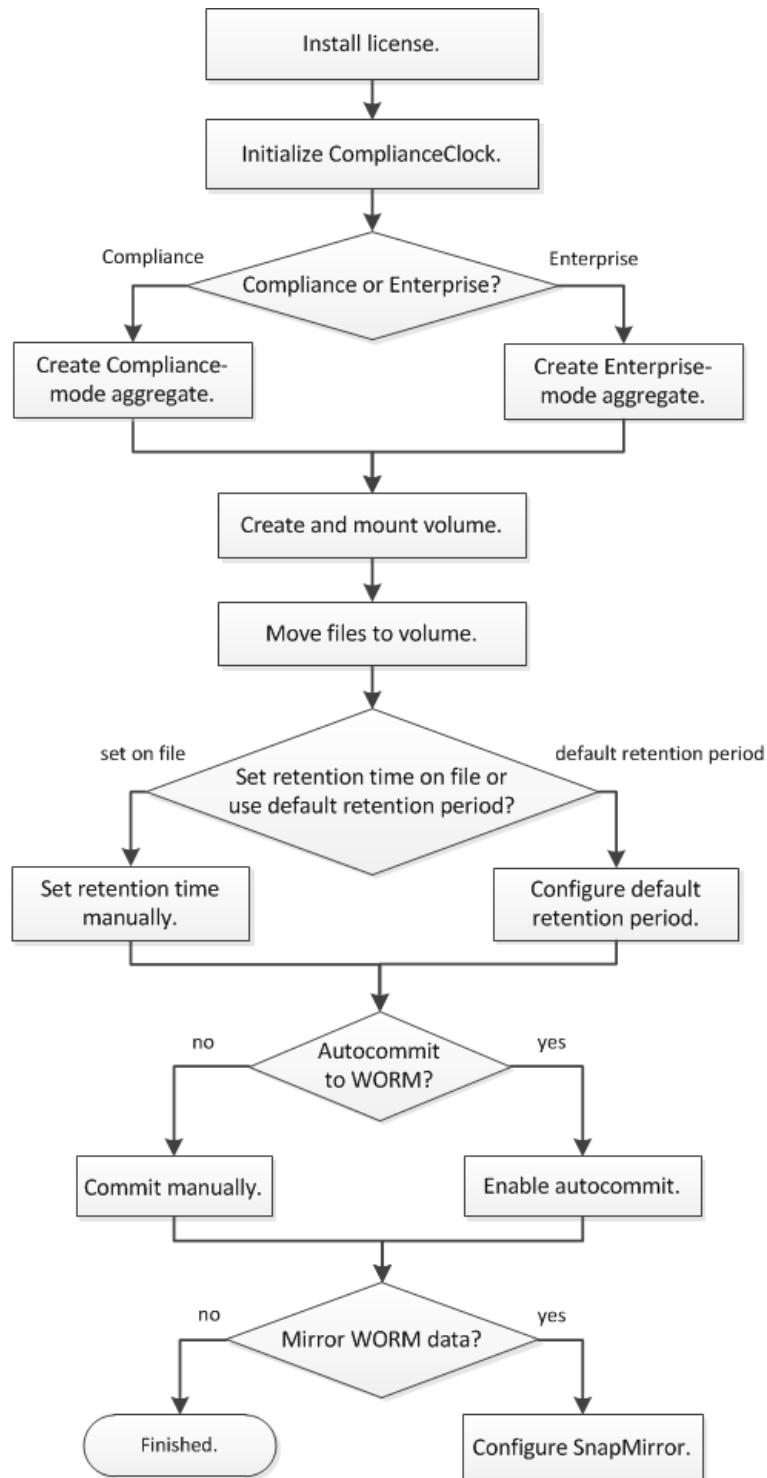
ONTAP offers both software- and hardware-based encryption technologies for ensuring that data at rest cannot be read if the storage medium is repurposed, returned, misplaced, or stolen.

Disclaimer: NetApp cannot guarantee that SnapLock-protected WORM files on self-encrypting drives or volumes will be retrievable if the authentication key is lost or if the number of failed authentication attempts exceeds the specified limit and results in the drive being permanently locked. You are responsible for ensuring against authentication failures.

Note: Starting with ONTAP 9.2, encrypted volumes are supported on SnapLock aggregates.

SnapLock workflow

You specify which SnapLock mode you want to use, Compliance or Enterprise, when you create a SnapLock aggregate. The SnapLock volume you create inherits the setting. You typically use a file archiving application to move files from primary storage to the SnapLock volume.



Configuring SnapLock

You must install the SnapLock license, initialize the ComplianceClock, create and configure a SnapLock aggregate and volume, and mount the volume before you can commit files to WORM.

Steps

1. [Installing the license](#) on page 8
2. [Initializing the ComplianceClock](#) on page 8
3. [Creating a SnapLock aggregate](#) on page 9
4. [Creating a SnapLock volume](#) on page 10
5. [Mounting a SnapLock volume](#) on page 11
6. [Setting the retention time](#) on page 11
7. [Verifying SnapLock settings](#) on page 14

Installing the license

A SnapLock license entitles you to use both SnapLock Compliance mode and SnapLock Enterprise mode. SnapLock licenses are issued on a per-node basis. You must install a license for each node that hosts a SnapLock aggregate.

Before you begin

You must be a cluster administrator to perform this task.

About this task

You should have received the SnapLock license keys from your sales representative.

Steps

1. Install the SnapLock license for a node:

```
system license add -license-code license_key
```

Example

The following command installs the license with the key **AAAAAAAAAAAAAAAAAAAAAAAAAAAAA**.

```
cluster1::> system license add -license-code AAAAAAAAAAAAAAAAAAAAAAAAAAAAA
```

2. Repeat the previous step for each node license.

Initializing the ComplianceClock

The SnapLock ComplianceClock ensures against tampering that might alter the retention period for WORM files. You must initialize the *system ComplianceClock* on each node that hosts a SnapLock aggregate. Once you initialize the ComplianceClock on a node, you cannot initialize it again.

Before you begin

- You must be a cluster administrator to perform this task.
- The SnapLock license must be installed on the node.

About this task

The time on the system ComplianceClock is inherited by the *volume ComplianceClock*, which controls the retention period for WORM files on the volume. The volume ComplianceClock is initialized automatically when you create a new SnapLock volume.

Important: The initial setting of the ComplianceClock is based on the current system clock. For that reason, you should verify that the system time and time zone are correct before initializing the ComplianceClock. Once you initialize the ComplianceClock on a node, you cannot initialize it again.

Steps

1. Initialize the system ComplianceClock:

```
snaplock compliance-clock initialize -node node_name
```

Example

The following command initializes the system ComplianceClock on **node1**:

```
cluster1::> snaplock compliance-clock initialize -node node1
```

2. When prompted, confirm that the system clock is correct and that you want to initialize the ComplianceClock:

Example

```
Warning: You are about to initialize the secure ComplianceClock of
the node "node1" to the current value of the node's system clock.
This procedure can be performed only once on a given node, so you
should ensure that the system time is set correctly before proceeding.
```

```
The current node's system clock is: Mon Apr 25 06:04:10 GMT 2016
```

```
Do you want to continue? (y|n): y
```

3. Repeat this procedure for each node that hosts a SnapLock aggregate.

Creating a SnapLock aggregate

You must create a SnapLock aggregate before creating a SnapLock volume. The SnapLock mode for the aggregate—Compliance or Enterprise—is inherited by the volumes in the aggregate.

Before you begin

- You must be a cluster administrator to perform this task.
- The SnapLock license must be installed on the node.
- The ComplianceClock on the node must be initialized.
- If you have partitioned the disks as “root”, “data1”, and “data2”, you must ensure that spare disks are available.

About this task

- You cannot create Compliance aggregates for array LUNs.
- You cannot create Compliance aggregates with the SyncMirror option.

You can destroy or rename an Enterprise aggregate at any time. You cannot destroy a Compliance aggregate until the retention period has elapsed. You can never rename a Compliance aggregate.

Step

1. Create a SnapLock aggregate:

```
storage aggregate create -aggregate aggregate_name -node node_name -
diskcount number_of_disks -snaplock-type compliance|enterprise
```

For a complete list of options, see the man page for the command.

Example

The following command creates a SnapLock **compliance** aggregate named **aggr1** with three disks on **node1**:

```
cluster1::> storage aggregate create -aggregate aggr1 -node node1 -
diskcount 3 -snaplock-type compliance
```

Creating a SnapLock volume

You must create a SnapLock volume for the files or Snapshot copies that you want to commit to the WORM state. The volume inherits the SnapLock mode—Compliance or Enterprise—from the SnapLock aggregate, and the ComplianceClock time from the node.

Before you begin

- The SnapLock aggregate must be online.
- You must have created a standard aggregate and a SVM to host the SnapLock volume.

About this task

You can destroy or rename an Enterprise volume at any time. You cannot destroy a Compliance volume until the retention period has elapsed. You can never rename a Compliance volume.

You can clone SnapLock volumes, but you cannot clone the files that are on the SnapLock volumes.

Step

1. Create a SnapLock volume:

```
volume create -vserver SVM_name -volume volume_name -aggregate
aggregate_name
```

For a complete list of options, see the man page for the command. The following options are not available for SnapLock volumes: `-nvfail`, `-atime-update`, `-is-autobalance-eligible`, `-space-mgmt-try-first`, and `vmalign`.

Example

The following command creates a SnapLock **compliance** volume named **vol1** on **aggr1** on **vs1**:

```
cluster1::> volume create -vserver vs1 -volume vol1 -aggregate aggr1
```

Mounting a SnapLock volume

You can mount a SnapLock volume to a junction path in the SVM namespace for NAS client access.

Before you begin

The SnapLock volume must be online.

About this task

- You can mount a SnapLock volume only under the root of the SVM.
- You cannot mount a regular volume under a SnapLock volume.

Step

1. Mount a SnapLock volume:

```
volume mount -vserver SVM_name -volume volume_name -junction-path path
```

For a complete list of options, see the man page for the command.

Example

The following command mounts a SnapLock volume named `vol1` to the junction path `/sales` in the `vs1` namespace:

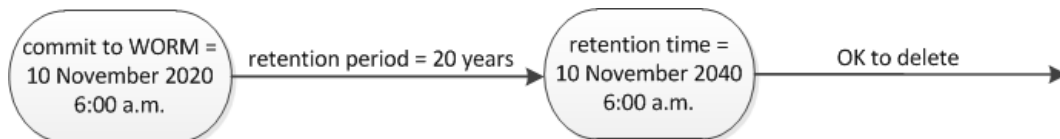
```
cluster1::> volume mount -vserver vs1 -volume vol1 -junction-path /
sales
```

Setting the retention time

You can set the retention time for a file explicitly, or you can use the default retention period for the volume to derive the retention time. Unless you set the retention time explicitly, SnapLock uses the default retention period to calculate the retention time.

Understanding retention period and retention time

The *retention period* for a WORM file specifies the length of time the file must be retained after it is committed to the WORM state. The *retention time* for a WORM file is the time after which the file no longer needs to be retained. A retention period of 20 years for a file committed to the WORM state on 10 November 2020 6:00 a.m., for example, would yield a retention time of 10 November 2040 6:00 a.m.



Understanding the default retention period

A SnapLock Compliance or Enterprise volume has three retention period values:

- Minimum retention period (**min**), with a default value of 0
- Maximum retention period (**max**), with a default value of 30 years

- Default retention period, with a default value that depends on the mode:
 - For Compliance mode, the default is equal to **max**.
 - For Enterprise mode, the default is equal to **min**.

So, if you do not set the retention time explicitly before committing a Compliance-mode file to the WORM state, and you do not modify the defaults, the file will be retained for 30 years. By the same token, if you do not set the retention time explicitly before committing an Enterprise-mode file to the WORM state, and you do not modify the defaults, the file will be retained for 0 years, effectively, not at all.

Choices

- [Setting the retention time for a file explicitly](#) on page 12
- [Setting the default retention period](#) on page 12

Setting the retention time for a file explicitly

You can set the retention time for a file explicitly by modifying its last access time. You can use any suitable command or program over NFS or CIFS to modify the last access time.

About this task

After a file has been committed to WORM, you can extend but not shorten the retention time. The retention time is stored in the `atime` field for the file.

Note: You cannot explicitly set the retention time of a file to **infinite**. That value is only available when you use the default retention period to calculate the retention time.

Step

1. Use a suitable command or program to modify the last access time for the file whose retention time you want to set.

Example

In a UNIX shell, use the following command to set a retention time of 21 November 2020 6:00 a.m. on a file named `document.txt`:

```
touch -a -t 202011210600 document.txt
```

Example

In PowerShell, use the following command to set a retention time of 21 November 2020 6:00 a.m. on a file named `document.txt`:

```
$(Get-Item document.txt).lastaccesstime=$(Get-Date "11/21/2020 06:00 am")
```

Setting the default retention period

You can use the `volume snaplock modify` command to set the default retention period for files on a SnapLock volume.

Before you begin

The SnapLock volume must be online.

About this task

The following table shows the possible values for the default retention period option:

Note: The default retention period must be greater than or equal to (\geq) the minimum retention period and less than or equal to (\leq) the maximum retention period.

Value	Unit	Notes
0 - 65535	seconds	
0 - 24	hours	
0 - 365	days	
0 - 12	months	
0 - 70	years	
max	-	Use the maximum retention period.
min	-	Use the minimum retention period.
infinite	-	Retain the files forever.

The values and ranges for the maximum and minimum retention periods are identical, except for **max** and **min**, which are not applicable. For more information about this task, see “Understanding the default retention period.”

[Understanding the default retention period](#) on page 11

You can use the `volume snaplock show` command to view the retention period settings for the volume. For more information, see the man page for the command.

Note: After a file has been committed to the WORM state, you can extend but not shorten the retention period.

Step

1. Set the default retention period for files on a SnapLock volume:

```
volume snaplock modify -vserver SVM_name -volume volume_name -default-
retention-period default_retention_period -minimum-retention-period
min_retention_period -maximum-retention-period max_retention_period
```

For a complete list of options, see the man page for the command.

Note: The following examples assume that the minimum and maximum retention periods have not been modified previously.

Example

The following command sets the default retention period for a Compliance or Enterprise volume to 20 days:

```
cluster1::> volume snaplock modify -vserver vs1 -volume vol1 -default-
retention-period 20days
```

Example

The following command sets the default retention period for a Compliance volume to 70 years:

```
cluster1::> volume snaplock modify -vserver vs1 -volume vol1 -maximum-
retention-period 70years
```

Example

The following command sets the default retention period for an Enterprise volume to 10 years:

```
cluster1::> volume snaplock modify -vserver vs1 -volume voll -default-
retention-period max -maximum-retention-period 10years
```

Example

The following commands set the default retention period for an Enterprise volume to 10 days:

```
cluster1::> volume snaplock modify -vserver vs1 -volume voll -minimum-
retention-period 10days
cluster1::> volume snaplock modify -vserver vs1 -volume voll -default-
retention-period min
```

Example

The following command sets the default retention period for a Compliance volume to infinite:

```
cluster1::> volume snaplock modify -vserver vs1 -volume voll -default-
retention-period infinite -maximum-retention-period infinite
```

Verifying SnapLock settings

You can use the `volume file fingerprint start` and `volume file fingerprint dump` commands to view key information about files and volumes, including the file type (regular, WORM, or WORM appendable), the volume expiration date, and so forth.

Steps

1. Generate a file fingerprint:

```
volume file fingerprint start -vserver SVM_name -file file_path
```

Example

```
svml1::> volume file fingerprint start -vserver svml -
file /vol/sle/vol/fl
File fingerprint operation is queued. Run "volume file fingerprint
show -session-id 16842791" to view the fingerprint session status.
```

The command generates a session ID that you can use as input to the `volume file fingerprint dump` command.

Note: You can use the `volume file fingerprint show` command with the session ID to monitor the progress of the fingerprint operation. Make sure that the operation has completed before attempting to display the fingerprint.

2. Display the fingerprint for the file:

```
volume file fingerprint dump -session-id session_ID
```

Example

```
svml1::> volume file fingerprint dump -session-id 33619976
Vserver:vs_kdivya_vsiml
Session-ID:33619976
Volume:slc_vol
Path:/vol/slc_vol/file1
```

```

Data Fingerprint:M0FJVevxNSJm3C/4Bn5oEEYH51CrudOzZYK4r5Cfylg=
Metadata Fingerprint:8iMjqJXiNcggXT5XuRhLiEwIrJEihDmwS0hrexnjgmc=
Fingerprint Algorithm:SHA256
Fingerprint Scope:data-and-metadata
Fingerprint Start Time:1460612586
Formatted Fingerprint Start Time:Thu Apr 14 05:43:06 GMT 2016
Fingerprint Version:3
SnapLock License:available
Vserver UUID:acf7ae64-00d6-11e6-a027-0050569c55ae
Volume MSID:2152884007
Volume DSID:1028
Hostname:kdivya-vsimg
Filer ID:5f18eda2-00b0-11e6-914e-6fb45e537b8d
Volume Containing Aggregate:slc_aggr1
Aggregate ID:c84634aa-c757-4b98-8f07-eefe32565f67
SnapLock System ComplianceClock:1460610635
Formatted SnapLock System ComplianceClock:Thu Apr 14 05:10:35 GMT 2016
Volume SnapLock Type:compliance
Volume ComplianceClock:1460610635
Formatted Volume ComplianceClock:Thu Apr 14 05:10:35 GMT 2016
Volume Expiry Date:1465880998
Is Volume Expiry Date Wraparound:false
Formatted Volume Expiry Date:Tue Jun 14 05:09:58 GMT 2016
Filesystem ID:1028
File ID:96
File Type:worm
File Size:1048576
Creation Time:1460612515
Formatted Creation Time:Thu Apr 14 05:41:55 GMT 2016
Modification Time:1460612515
Formatted Modification Time:Thu Apr 14 05:41:55 GMT 2016
Changed Time:1460610598
Is Changed Time Wraparound:false
Formatted Changed Time:Thu Apr 14 05:09:58 GMT 2016
Retention Time:1465880998
Is Retention Time Wraparound:false
Formatted Retention Time:Tue Jun 14 05:09:58 GMT 2016
Access Time:-
Formatted Access Time:-
Owner ID:0
Group ID:0
Owner SID:-
Fingerprint End Time:1460612586
Formatted Fingerprint End Time:Thu Apr 14 05:43:06 GMT 2016

```

Committing files to WORM

You can use a command or program to commit files to WORM manually, or use the SnapLock autocommit feature to commit files to WORM automatically.

Choices

- [Committing files to WORM manually](#) on page 16
- [Autocommitting files to WORM](#) on page 16
- [Creating a WORM appendable file](#) on page 17

Committing files to WORM manually

You commit a file to WORM manually by making the file read-only. You can use any suitable command or program over NFS or CIFS to change the read-write attribute of a file to read-only.

Before you begin

- The file you want to commit must reside on a SnapLock volume.
- The file must be writable.

About this task

The volume ComplianceClock time is written to the `ctime` field of the file when the command or program is executed. The ComplianceClock time determines when the retention time for the file has been reached.

Step

1. Use a suitable command or program to change the read-write attribute of a file to read-only.

Example

In a UNIX shell, use the following command to make a file named `document.txt` read-only:

```
chmod -w document.txt
```

Example

In a Windows shell, use the following command to make a file named `document.txt` read-only:

```
attrib +r document.txt
```

Autocommitting files to WORM

The SnapLock autocommit feature enables you to commit files to WORM automatically.

Before you begin

- The files you want to autocommit must reside on a SnapLock volume.
- The SnapLock volume must be online.

Note: The SnapLock autocommit feature scans through all of the files in the volume and commits a file if it meets the autocommit requirement. There might be a time interval between when the file is ready for autocommit and when it is actually committed by the SnapLock autocommit scanner. However, the file is still protected from modifications and deletion by the file system as soon as it is eligible for autocommit.

About this task

The *autocommit period* specifies the amount of time that files must remain unchanged before they are autocommitted. Changing a file before the autocommit period has elapsed restarts the autocommit period for the file.

The following table shows the possible values for the autocommit period:

Value	Unit	Notes
none	-	The default.
2 to 24	hours	0 or 1 hour are invalid entries.
1 to 7	days	-

Step

1. Autocommit files on a SnapLock volume to WORM:

```
volume snaplock modify -vserver SVM_name -volume volume_name -
autocommit-period autocommit_period
```

For a complete list of options, see the man page for the command.

Example

The following command autocommits the files on volume `vol1` of SVM `vs1`, as long as the files remain unchanged for 5 hours:

```
cluster1::>volume snaplock modify -vserver vs1 -volume vol1 -
autocommit-period 5hours
```

Creating a WORM appendable file

You can use a WORM appendable file to retain data that is written incrementally, like log entries. Data is appended to the file in 256 KB chunks. As each chunk is written, the previous chunk becomes WORM-protected. You cannot delete the file until the retention period has elapsed.

Before you begin

The WORM appendable file must reside on a SnapLock volume.

About this task

Data does not have to be written sequentially to the active 256 KB chunk. When data is written to byte $n \times 256KB + 1$ of the file, the previous 256 KB segment becomes WORM-protected.

Steps

1. Use a suitable command or program to create a zero-length file with the desired retention time.

Example

In a UNIX shell, use the following command to set a retention time of 21 November 2020 6:00 a.m. on a zero-length file named `document.txt`:

```
touch -a -t 202011210600 document.txt
```

2. Use a suitable command or program to change the read-write attribute of the file to read-only.

Example

In a UNIX shell, use the following command to make a file named `document.txt` read-only:

```
chmod 444 document.txt
```

3. Use a suitable command or program to change the read-write attribute of the file back to writable.

Note: This step is not deemed a compliance risk because there is no data in the file.

Example

In a UNIX shell, use the following command to make a file named `document.txt` writable:

```
chmod 777 document.txt
```

4. Use a suitable command or program to start writing data to the file.

Example

In a UNIX shell, use the following command to write data to `document.txt`:

```
echo test data >> document.txt
```

Note: Change the file permissions back to read-only when you no longer need to append data to the file.

Using Volume Append Mode to create WORM appendable files

Because a file on a SnapLock volume cannot be deleted until the file retention time has expired, you may want to create a WORM appendable file for logging purposes. Beginning in ONTAP 9.3, you can create WORM appendable files by default on a SnapLock volume using Volume Append Mode (VAM).

Before you begin

The SnapLock Volume must be unmounted and empty of Snapshot copies and user-created files.

About this task

Use the `volume snaplock modify` command to configure VAM.

Note: VAM and audit logging are not supported on the same volume.

Steps

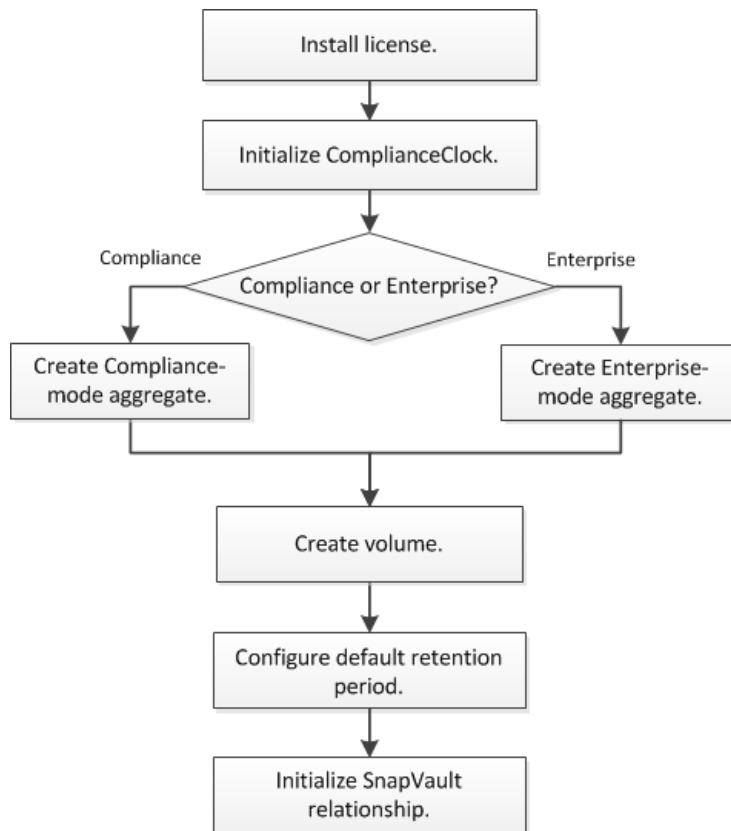
1. Enable VAM:

```
volume snaplock modify -vserver vs1 -volume vol1 -is-volume-append-mode-enabled true
```

2. Create files with write permissions.
The files are WORM appendable by default.

Committing Snapshot copies to WORM

You can use SnapLock for SnapVault to WORM-protect Snapshot copies on secondary storage. You perform all of the basic SnapLock tasks on the SnapVault destination. The destination volume is automatically mounted read-only, so there is no need to explicitly commit the Snapshot copies to WORM.



Before you begin

- The source cluster must be running ONTAP 8.2.2 or later.
- The source and destination aggregates must be 64-bit.
- The source volume cannot be a SnapLock volume.
- The source and destination volumes must be created in peered clusters with peered SVMs. For more information, see the *Cluster Peering Express Guide*.
- If volume autogrow is disabled, the free space on the destination volume must be at least five percent more than the used space on the source volume.

About this task

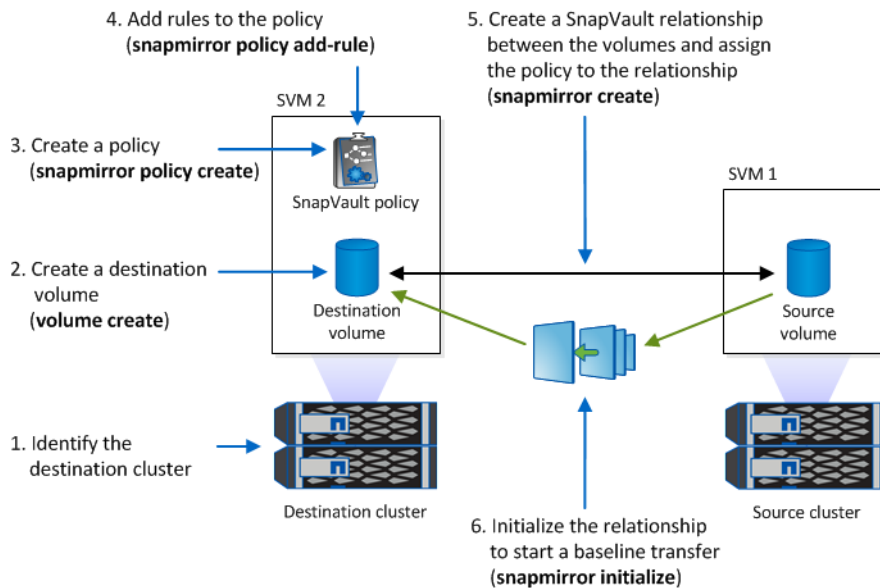
The source volume can use NetApp or non-NetApp storage. For non-NetApp storage, you must use FlexArray Virtualization.

Note: You cannot rename a Snapshot copy that is committed to the WORM state.

For MetroCluster configurations, you should be aware of the following:

- You can create a SnapVault relationship only between sync-source SVMs, not between a sync-source SVM and a sync-destination SVM.
- You can create a SnapVault relationship from a volume on a sync-source SVM to a data-serving SVM.
- You can create a SnapVault relationship from a volume on a data-serving SVM to a DP volume on a sync-source SVM.

The following illustration shows the procedure for initializing a SnapVault relationship:



Steps

1. Identify the destination cluster.
2. On the destination cluster, install the SnapLock license, initialize the ComplianceClock, and create a SnapLock aggregate, as described in [Configuring SnapLock](#) on page 8.
3. On the destination cluster, create a SnapLock destination volume of type **DP** that is either the same or greater in size than the source volume:

```
volume create -vserver SVM_name -volume volume_name -aggregate
aggregate_name -type DP -size size
```

Note: The SnapLock mode, Compliance or Enterprise, is inherited from the aggregate. Version-flexible destination volumes are not supported. The language setting of the destination volume must match the language setting of the source volume.

Example

The following command creates a 2 GB SnapLock **Compliance** volume named **dstvol1B** in **SVM2** on the aggregate **node01_aggr**:

```
cluster2::> volume create -vserver SVM2 -volume dstvol1B -aggregate
node01_aggr -type DP -size 2GB
```

4. On the destination cluster, set the default retention period, as described in [Setting the default retention period](#) on page 12.

You can use the `snapshot extend-snapshot-retention` command to extend the retention period for Snapshot copies committed to WORM. For more information, see the man page for the command.

5. On the destination SVM, create a SnapVault policy:

```
snapmirror policy create -vserver SVM_name -policy policy_name -type vault
```

Example

The following command creates the SVM-wide SnapVault policy `SVM1-vault`:

```
SVM2::> snapmirror policy create -vserver SVM2 -policy SVM1-vault -type vault
```

6. Add rules to the policy that define Snapshot copy labels and the retention policy for each label:

```
snapmirror policy add-rule -vserver SVM_name -policy policy_name -snapmirror-label label -keep number_of_snapshot_copies_to_retain
```

The following labels are available:

- Daily
- Weekly
- Monthly
- Hourly
- Unlimited

Note: The labels are case-sensitive.

Example

The following command adds a rule to the `SVM1-vault` policy that defines the `Daily` label and specifies that 30 Snapshot copies matching the label should be kept in the vault:

```
SVM2::> snapmirror policy add-rule -vserver SVM2 -policy SVM1-vault -snapmirror-label Daily -keep 30
```

7. On the destination SVM, create a SnapVault schedule:

```
job schedule cron create -name schedule_name -dayofweek day_of_week -hour hour -minute minute
```

Example

The following command creates a SnapVault schedule named `weekendcron`:

```
SVM2::> job schedule cron create -name weekendcron -dayofweek "Saturday, Sunday" -hour 3 -minute 0
```

8. On the destination SVM, create the SnapVault relationship and assign the SnapVault policy and schedule:

```
snapmirror create -source-path source_path -destination-path destination_path -type XDP -policy policy_name -schedule schedule_name
```

Example

The following command creates a SnapVault relationship between the source volume `srcvolA` on `SVM1` and the destination volume `dstvolB` on `SVM2`, and assigns the policy `SVM1-vault` and the schedule `weekendcron`:

```
SVM2::> snapmirror create -source-path SVM1:srcvolA -destination-path  
SVM2:dstvolB -type XDP -policy SVM1-vault -schedule weekendcron
```

9. On the destination SVM, initialize the SnapVault relationship:

```
snapmirror initialize -destination-path destination_path
```

The initialization process performs a *baseline transfer* to the destination volume. SnapMirror makes a Snapshot copy of the source volume, then transfers the copy and all of the data blocks it references to the destination volume.

Example

The following command initializes the relationship between the source volume **srcvolA** on **SVM1** and the destination volume **dstvolB** on **SVM2**:

```
SVM2::> snapmirror initialize -destination-path SVM2:dstvolB
```

Related information

[Cluster peering express configuration](#)

[Volume express backup using SnapVault](#)

Mirroring WORM files

You can use SnapMirror to replicate WORM files to another geographic location for disaster recovery and other purposes. Both the source volume and destination volume must be configured for SnapLock, and both volumes must have the same SnapLock mode, Compliance or Enterprise. All key SnapLock properties of the volume and files are replicated.

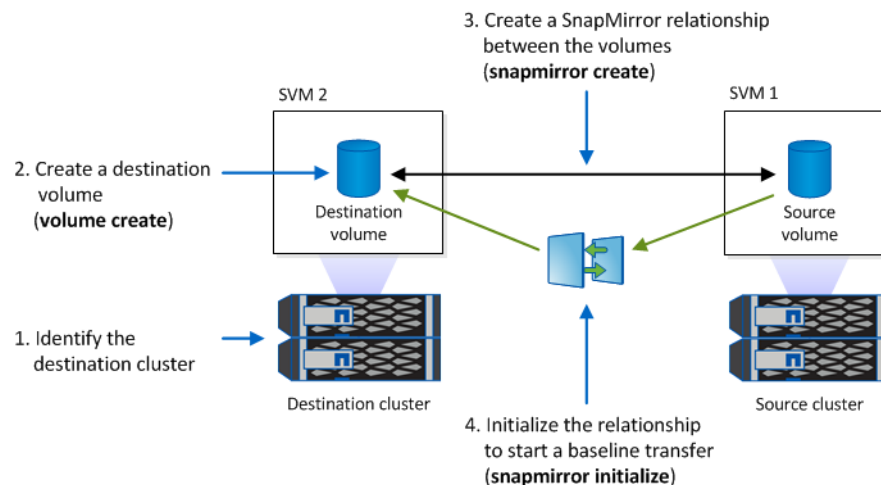
Before you begin

The source and destination volumes must be created in peered clusters with peered SVMs. For more information, see the *Cluster Peering Express Guide* and the *SVM Disaster Recovery Preparation Express Guide*.

About this task

- A resync operation fails for a Compliance-mode volume if SnapLock determines that it will result in a loss of data. If a resync operation fails, you can use the `volume clone create` command to make a clone of the destination volume. You can then resync the source volume with the clone.
- SVM data protection relationships are not supported.
- Load-sharing data protection relationships are not supported.

The following illustration shows the procedure for initializing a SnapMirror relationship:



Steps

1. Identify the destination cluster.
2. On the destination cluster, install the SnapLock license, initialize the ComplianceClock, and create a SnapLock aggregate.

[Configuring SnapLock](#) on page 8

3. On the destination cluster, create a SnapLock destination volume of type **DP** that is either the same size as or greater in size than the source volume:

```
volume create -vserver SVM_name -volume volume_name -aggregate
aggregate_name -type DP -size size
```


Note: The SnapLock mode—Compliance or Enterprise—is inherited from the volume. Version-flexible destination volumes are not supported. The language setting of the destination volume must match the language setting of the source volume.

Example

The following command creates a 2 GB SnapLock **Compliance** volume named **dstvolB** in **SVM2** on the aggregate **node01_aggr**:

```
cluster2::> volume create -vserver SVM2 -volume dstvolB -aggregate
node01_aggr -type DP -size 2GB
```

4. On the destination SVM, create a SnapMirror policy:

```
snapmirror policy create -vserver SVM_name -policy policy_name
```

Example

The following command creates the SVM-wide policy **SVM1-mirror**:

```
SVM2::> snapmirror policy create -vserver SVM2 -policy SVM1-mirror
```

5. On the destination SVM, create a SnapMirror schedule:

```
job schedule cron create -name schedule_name -dayofweek day_of_week -
hour hour -minute minute
```

Example

The following command creates a SnapMirror schedule named **weekendcron**:

```
SVM2::> job schedule cron create -name weekendcron -dayofweek
"Saturday, Sunday" -hour 3 -minute 0
```

6. On the destination SVM, create a SnapMirror relationship:

```
snapmirror create -source-path source_path -destination-path
destination_path -type DP -policy policy_name -schedule schedule_name
```

Example

The following command creates a SnapMirror relationship between the source volume **srcvolA** on **SVM1** and the destination volume **dstvolB** on **SVM2**, and assigns the policy **SVM1-mirror** and the schedule **weekendcron**:

```
SVM2::> snapmirror create -source-path SVM1:srcvolA -destination-path
SVM2:dstvolB -type DP -policy SVM1-mirror -schedule weekendcron
```

7. On the destination SVM, initialize the SnapMirror relationship:

```
snapmirror initialize -destination-path destination_path
```

The initialization process performs a *baseline transfer* to the destination volume. SnapMirror makes a Snapshot copy of the source volume, then transfers the copy and all the data blocks that it references to the destination volume. It also transfers any other Snapshot copies on the source volume to the destination volume.

Example

The following command initializes the relationship between the source volume **srcvolA** on **SVM1** and the destination volume **dstvolB** on **SVM2**:

```
SVM2::> snapmirror initialize -destination-path SVM2:dstvolB
```

Related information

[Cluster peering express configuration](#)

[SVM disaster recovery express preparation](#)

[Volume disaster recovery express preparation](#)

Using the privileged delete feature

You can use an audited *privileged delete* to delete Enterprise-mode WORM files during the retention period. You must be a compliance administrator to perform a privileged delete.

Steps

1. [Creating an audit log](#) on page 27
2. [Creating a compliance administrator account](#) on page 28
3. [Enabling the privileged delete feature](#) on page 29
4. [Deleting WORM files using privileged delete](#) on page 29

Creating an audit log

You must create a SnapLock audit log before performing a privileged delete. The audit log records the creation and deletion of compliance administrator accounts, modifications to the log volume, whether privileged delete is enabled, and privileged delete operations.

Before you begin

You must be a cluster administrator to create a SnapLock aggregate.

About this task

You cannot delete an audit log until the log file retention period has elapsed. You cannot modify an audit log even after the retention period has elapsed.

Important: SnapLock Enterprise volumes do not support audit logging. You must use a SnapLock Compliance volume for audit logging. The audit log volume must be mounted at the junction path `/snaplock_audit_log`. No other volume can use this junction path.

You can find the SnapLock audit logs in the `/snaplock_log` directory under the root of the audit log volume, in subdirectories named `privdel_log` (privileged delete operations) and `system_log` (everything else). Audit log file names contain the timestamp of the first logged operation, making it easy to search for records by the approximate time that operations were executed.

- You can use the `snaplock log file show` command to view the log files on the audit log volume.
- You can use the `snaplock log file archive` command to archive the current log file and create a new one, which is useful in cases where you need to record audit log information in a separate file.

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

Note: A data protection volume cannot be used as a SnapLock audit log volume.

Steps

1. Create a SnapLock Compliance aggregate.
[Creating a SnapLock aggregate](#) on page 9
2. On the SVM that you want to configure for audit logging, create a SnapLock Compliance volume.
[Creating a SnapLock volume](#) on page 10

3. Configure the SVM for audit logging using the Compliance volume:

```
snaplock log create -vserver SVM_name -volume volume_name -max-file-size
size -retention-period default_retention_period
```

Note: The minimum default retention period for audit log files is six months. If the retention period of an affected file is longer than the retention period of the audit log, the retention period of the log inherits the retention period of the file. So, if the retention period for a file deleted using privileged delete is 10 months, and the retention period of the audit log is 8 months, the retention period of the log is extended to 10 months.

Example

The following command configures the SVM **svM1** for audit logging using the Compliance volume **logVol1**. The audit log has a maximum size of 20 GB and is retained for eight months.

```
SVM1::> snaplock log create -vserver SVM1 -volume logVol1 -max-file-size
20GB -retention-period 8months
```

4. On the SVM that you configured for audit logging, mount the SnapLock Compliance volume at the junction path `/snaplock_audit_log`.

[Mounting a SnapLock volume](#) on page 11

Creating a compliance administrator account

You must have compliance administrator privileges to perform a privileged delete. These privileges are defined in the `vsadmin-snaplock` role. If you have not already been assigned that role, you can ask your cluster administrator to create an SVM administrator account with the compliance administrator role.

Before you begin

- You must be a cluster administrator to perform this task.
- You must have logged in on a secure connection (SSH, console, or ZAPI).

Step

1. Create an SVM administrator account with the compliance administrator role:

```
security login create -vserver SVM_name -user-or-group-name
user_or_group_name -application application -authmethod
authentication_method -role role -comment comment
```

Example

The following command enables the SVM administrator account **SnapLockAdmin** with the predefined `vsadmin-snaplock` role to access the SVM **svM1** using a password:

```
cluster1::> security login create -vserver SVM1 -user-or-group-name
SnapLockAdmin -application ssh -authmethod password -role vsadmin-
snaplock
```

Enabling the privileged delete feature

You must explicitly enable the privileged delete feature on the Enterprise volume that contains the WORM files you want to delete.

About this task

The value of the `-privileged-delete` option determines whether privileged delete is enabled. Possible values are **enabled**, **disabled**, and **permanently-disabled**.

Note: **permanently-disabled** is the terminal state. You cannot enable privileged delete on the volume after you set the state to **permanently-disabled**.

Step

1. Enable privileged delete for a SnapLock Enterprise volume:

```
volume snaplock modify -vserver SVM_name -volume volume_name -
privileged-delete disabled|enabled|permanently-disabled
```

Example

The following command enables the privileged delete feature for the Enterprise volume **dataVol** on the SVM **SVM1**:

```
SVM1::> volume snaplock modify -vserver SVM1 -volume dataVol -
privileged-delete enabled
```

Deleting WORM files using privileged delete

You can use the privileged delete feature to delete Enterprise-mode WORM files during the retention period.

Before you begin

- You must be a compliance administrator to perform this task.
- You must have created a SnapLock audit log and enabled the privileged delete feature on the Enterprise volume.

About this task

You cannot use a privileged delete operation to delete an expired WORM file. You can use the `volume file retention show` command to view the retention time of the WORM file that you want to delete. For more information, see the man page for the command.

Step

1. Delete a WORM file on an Enterprise volume:

```
volume file privileged-delete -vserver SVM_name -file file_path
```

Example

The following command deletes the file `/vol/dataVol/f1` on the SVM **SVM1**:

```
SVM1::> volume file privileged-delete -file /vol/dataVol/f1
```

Holding tamper-proof files for an indefinite period

Beginning in ONTAP 9.3, you can hold files in a tamper-proof state for the period of a litigation in SnapLock Compliance volumes using the Legal Hold feature.

About this task

A file under Legal Hold will behave like a WORM file with indefinite retention. The number of files that can be placed under Legal Hold depends on the space available on the volume.

Step

1. Manage the Legal Hold feature

If you want to...	Use this command...
Start an operation to place files under legal-hold.	<code>snaplock legal-hold begin -litigation-name -volume -path</code>
Start an operation to release legal-hold on files.	<code>snaplock legal-hold end -litigation-name -volume -path</code>
Show the progress of a legal-hold operation.	<code>snaplock legal-hold show</code>
Abort legal-hold operations	<code>snaplock legal-hold abort -operations-id</code>
Dump list of all litigations in a given Vserver to specified output path	<code>snaplock legal-hold dump-litigations -output-volume -output-directory-path [-volume]</code>
Dump list of all files held for litigation in a given volume to specified output path	<code>snaplock legal-hold dump-files -volume -litigation-name -output-volume -output-directory-path</code>

Using the Event Based Retention (EBR) feature

Beginning in ONTAP 9.3, you can use the Event Based Retention (EBR) feature on SnapLock Compliance volumes to retain files as defined by the event or policy.

Before you begin

EBR policies can only be created for data Vservers.

No policies or events will be created by default. It is necessary to create events or policies before applying them.

About this task

You can create EBR policies for a Vserver with a specific retention time. Apply this policy either on a single file or an entire directory. If applied on a directory, the EBR policy will be applied to all files inside that container.

If a file is not a WORM file, it will be committed to WORM state. If a file is a WORM file or a WORM appendable file with a retention period, its retention period will be extended by the retention period defined by the EBR policy.

Note: EBR policies cannot be applied to files under Legal Hold.

Steps

1. Implement the EBR policy.

If you want to...	Use this command...
Create an EBR policy	<code>snaplock event-retention policy create</code>
Modify an EBR policy	<code>snaplock event-retention policy modify</code>
Delete an EBR policy	<code>snaplock event-retention policy delete</code>
Show the progress of an EBR policy	<code>snaplock event-retention policy show</code>

2. Manage the EBR policy.

If you want to...	Use this command...
Apply an EBR policy on one or more files	<code>snaplock event-retention operation start</code>
Abort the EBR policy operation	<code>snaplock event-retention operation abort</code>
Check the status of the EBR policy operation	<code>snaplock event-retention operation show</code>

SnapLock APIs

You can use Zephyr APIs to integrate with SnapLock functionality in scripts or workflow automation. The APIs use XML messaging over HTTP, HTTPS, and Windows DCE/RPC.

file-fingerprint-abort

Abort a file fingerprint operation.

file-fingerprint-dump

Display file fingerprint information.

file-fingerprint-get-iter

Display the status of file fingerprint operations.

file-fingerprint-start

Generate a file fingerprint.

snaplock-archive-vserver-log

Archive the active audit log file.

snaplock-create-vserver-log

Create an audit log configuration for an SVM.

snaplock-delete-vserver-log

Delete an audit log configuration for an SVM.

snaplock-file-privileged-delete

Execute a privileged delete operation.

snaplock-get-file-retention

Get the retention period of a file.

snaplock-get-node-compliance-clock

Get the node ComplianceClock date and time.

snaplock-get-vserver-active-log-files-iter

Display the status of active log files.

snaplock-get-vserver-log-iter

Display the audit log configuration.

snaplock-modify-vserver-log

Modify the audit log configuration for an SVM.

snaplock-set-file-retention

Set the retention time for a file.

snaplock-set-node-compliance-clock

Set the node ComplianceClock date and time.

snaplock-volume-set-privileged-delete

Set the privileged-delete option on a SnapLock Enterprise volume.

volume-get-snaplock-attrs

Get the attributes of a SnapLock volume.

volume-set-snaplock-attrs

Set the attributes of a SnapLock volume.

Where to find additional information

You can learn more about the tasks described in this guide in NetApp's extensive documentation library.

- [*ONTAP 9 commands*](#)
Describes SnapLock commands in reference format.
- [*Cluster management using System Manager*](#)
Describes how to use OnCommand System Manager to install SnapLock licenses, set the Compliance Clock, create SnapLock aggregates and volumes, and configure SnapLock volumes.
- [*NetApp Documentation: OnCommand Workflow Automation \(current releases\)*](#)
Describes how to use the OnCommand Workflow Automation scripting tool to perform SnapLock-related tasks.
- [*Cluster peering express configuration*](#)
Describes how to quickly configure peer relationships between clusters and SVMs.
- [*Volume express backup using SnapVault*](#)
Describes how to prepare volumes for SnapVault replication.
- [*Volume disaster recovery express preparation*](#)
Describes how to prepare volumes for SnapMirror replication.
- [*Volume disaster express recovery*](#)
Describes how to restore volumes after a disaster.

Copyright information

Copyright © 1994–2017 NetApp, Inc. All rights reserved. Printed in the U.S.

No part of this document covered by copyright may be reproduced in any form or by any means—graphic, electronic, or mechanical, including photocopying, recording, taping, or storage in an electronic retrieval system—without prior written permission of the copyright owner.

Software derived from copyrighted NetApp material is subject to the following license and disclaimer:

THIS SOFTWARE IS PROVIDED BY NETAPP "AS IS" AND WITHOUT ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE, WHICH ARE HEREBY DISCLAIMED. IN NO EVENT SHALL NETAPP BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

NetApp reserves the right to change any products described herein at any time, and without notice. NetApp assumes no responsibility or liability arising from the use of products described herein, except as expressly agreed to in writing by NetApp. The use or purchase of this product does not convey a license under any patent rights, trademark rights, or any other intellectual property rights of NetApp.

The product described in this manual may be protected by one or more U.S. patents, foreign patents, or pending applications.

RESTRICTED RIGHTS LEGEND: Use, duplication, or disclosure by the government is subject to restrictions as set forth in subparagraph (c)(1)(ii) of the Rights in Technical Data and Computer Software clause at DFARS 252.277-7103 (October 1988) and FAR 52-227-19 (June 1987).

Trademark information

Active IQ, AltaVault, Arch Design, ASUP, AutoSupport, Campaign Express, Clustered Data ONTAP, Customer Fitness, Data ONTAP, DataMotion, Element, Fitness, Flash Accel, Flash Cache, Flash Pool, FlexArray, FlexCache, FlexClone, FlexPod, FlexScale, FlexShare, FlexVol, FPolicy, Fueled by SolidFire, GetSuccessful, Helix Design, LockVault, Manage ONTAP, MetroCluster, MultiStore, NetApp, NetApp Insight, OnCommand, ONTAP, ONTAPI, RAID DP, RAID-TEC, SANscreen, SANshare, SANtricity, SecureShare, Simplicity, Simulate ONTAP, Snap Creator, SnapCenter, SnapCopy, SnapDrive, SnapIntegrator, SnapLock, SnapManager, SnapMirror, SnapMover, SnapProtect, SnapRestore, Snapshot, SnapValidator, SnapVault, SolidFire, SolidFire Helix, StorageGRID, SyncMirror, Tech OnTap, Unbound Cloud, and WAFL and other names are trademarks or registered trademarks of NetApp, Inc., in the United States, and/or other countries. All other brands or products are trademarks or registered trademarks of their respective holders and should be treated as such. A current list of NetApp trademarks is available on the web.

<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*](mailto: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., 495 East Java Drive, Sunnyvale, CA 94089 U.S.
- Telephone: +1 (408) 822-6000
- Fax: +1 (408) 822-4501
- Support telephone: +1 (888) 463-8277

Index

7-Mode transition
SnapLock [5](#)

A

about this guide
 deciding whether to use the Archive and Compliance
 Using SnapLock Technology Power Guide [4](#)
accounts, SnapLock compliance administrator
 creating [28](#)
administrator accounts, SnapLock compliance
 creating [28](#)
APIs
 Data ONTAP [32](#)
 descriptions [32](#)
 SnapLock [32](#)
audience
 for the guide [4](#)
audit logs, SnapLock
 creating [27](#)
autocommitting
 WORM files [16](#)

C

comments
 how to send feedback about documentation [37](#)
compliance administrator accounts, SnapLock
 creating [28](#)
Compliance mode
 differences between Enterprise mode and [5](#)
Compliance volumes, SnapLock
 creating [10](#)
 mounting [11](#)
ComplianceClock
 initializing [8](#)
creating
 audit logs [27](#)
 SnapLock audit logs [27](#)
 SnapLock compliance administrator accounts [28](#)
 SnapLock Compliance aggregates [9](#)
 SnapLock Compliance volumes [10](#)
 SnapLock Enterprise aggregates [9](#)
 SnapLock Enterprise volumes [10](#)
creating a disaster recovery relationship
 SnapLock [24](#)

D

Data ONTAP APIs [32](#)
default retention periods
 setting [12](#)
 understanding SnapLock [11](#)
disaster recovery
 creating a mirror copy [24](#)
documentation
 additional information about SnapLock [34](#)

how to receive automatic notification of changes to
[37](#)
how to send feedback about [37](#)

E

enabling
 privileged delete [29](#)
 SnapLock privileged delete [29](#)
Enterprise mode
 differences between Compliance mode and [5](#)
Enterprise volumes, SnapLock
 creating [10](#)
 mounting [11](#)
Event Based Retention (EBR)
 using to manage SnapLock Compliance volumes [31](#)

F

feedback
 how to send comments about documentation [37](#)
files
 committing to WORM state [16](#)
 differences between SnapLock retention period and
 retention time [11](#)
 fingerprint [14](#)
 setting the retention time for, explicitly [12](#)
 verifying settings [14](#)
 verifying SnapLock settings [14](#)
fingerprint
 of a file [14](#)
FlexArray Virtualization
 use with SnapLock [5, 20](#)

I

information
 how to send feedback about improving
 documentation [37](#)

L

legal-hold, SnapLock
 managing [30](#)
licenses, SnapLock
 installing [8](#)
logs, SnapLock audit
 creating [27](#)

M

managing
 Legal Hold [30](#)
 SnapLock audit logs [30](#)
MetroCluster
 use with SnapLock [5, 20](#)
MetroCluster configurations
 use with SnapLock [4](#)

- mounting
 - SnapLock Compliance volumes [11](#)
 - SnapLock Enterprise volumes [11](#)
- N**
- nodes
 - installing the SnapLock license on [8](#)
- P**
- performing
 - privileged delete [29](#)
 - SnapLock privileged delete [29](#)
- periods, default retention
 - setting [12](#)
- Power Guides
 - requirements for using this guide [4](#)
- privileged delete
 - deleting WORM files by using [29](#)
- privileged delete, SnapLock
 - enabling [29](#)
 - performing [29](#)
- provisioning storage
 - creating SnapLock Compliance aggregates [9](#)
 - creating SnapLock Enterprise aggregates [9](#)
- R**
- retention periods
 - differences between retention time and [11](#)
- retention periods, default
 - setting [12](#)
- retention times
 - differences between retention period and [11](#)
 - setting explicitly for a file [12](#)
- S**
- SnapLock
 - 7-Mode transition [5](#)
 - compliance administrator [5](#)
 - Compliance and Enterprise modes [5](#)
 - creating a mirror copy [24](#)
 - differences between retention period and retention time [11](#)
 - disaster recovery [24](#)
 - initializing the ComplianceClock [8](#)
 - installing the license [8](#)
 - setting the retention period [12](#)
 - setting the retention time for a file explicitly [12](#)
 - SnapLock for SnapVault [5](#)
 - support for SnapMirror disaster recovery [5](#)
 - use with FlexArray Virtualization [5, 20](#)
 - use with MetroCluster [5, 20](#)
 - use with MetroCluster configurations [4](#)
 - where to find additional information about [34](#)
 - workflow for files [7](#)
 - workflow for Snapshot copies [20](#)
- SnapLock aggregates
 - creating [9](#)
- SnapLock APIs [32](#)
- SnapLock audit logs
 - creating [27](#)
- SnapLock compliance administrator accounts
 - creating [28](#)
- SnapLock Compliance volumes
 - managing using Event Based Retention (EBR) [31](#)
- SnapLock for SnapVault
 - protecting Snapshot copies [5](#)
- SnapLock Legal Hold
 - managing [30](#)
- SnapLock privileged delete
 - enabling [29](#)
 - performing [29](#)
- SnapLock volumes
 - creating [10](#)
 - creating WORM appendable files [18](#)
 - mounting [11](#)
- SnapMirror
 - creating a disaster recovery relationship [24](#)
 - replicating WORM files with SnapLock [5](#)
- Snapshot copies
 - committing to WORM [20](#)
 - WORM-protecting [5](#)
- storage provisioning
 - creating SnapLock aggregates [9](#)
- suggestions
 - how to send feedback about documentation [37](#)
- system ComplianceClock
 - initializing [8](#)
- T**
- Twitter
 - how to receive automatic notification of documentation changes [37](#)
- V**
- Volume Append Mode (VAM)
 - using to create WORM appendable files on SnapLock volumes [18](#)
- volumes
 - verifying settings [14](#)
- volumes, SnapLock
 - creating [10](#)
 - mounting [11](#)
- W**
- workflows
 - SnapLock [7](#)
- workflows, SnapLock
 - for Snapshot copies [20](#)
- WORM
 - committing files to [16](#)
 - committing Snapshot copies to [20](#)
 - creating an appendable file [17](#)
 - transitioning files to [16](#)
- WORM appendable file
 - creating [17](#)
- WORM appendable files
 - creating on SnapLock volumes [18](#)

WORM files

autocommitting [16](#)

committing with SnapLock [5](#)

mirroring [24](#)

replicating [24](#)

replicating with SnapMirror [5](#)

using the privileged delete feature to delete [29](#)