ONTAP Mediator documentation

ONTAP Mediator

NetApp
November 09, 2026

This PDF was generated from https://docs.netapp.com/us-en/ontap-mediator/index.html on November 09, 2026. Always check docs.netapp.com for the latest.

Table of Contents

ONTAP Mediator documentation . . . 1
Release notes . . . 2
What's new in ONTAP Mediator . . . 2
ONTAP Mediator support matrix . . . 5
OS support matrix . . . 5
SCST support matrix . . . 8
Get started . . . 10
Learn about ONTAP Mediator . . . 10
How ONTAP Mediator works . . . 10
ONTAP Mediator services . . . 10
Tools provided for the system administrator . . . 10
SCST and kernel updates . . . 11
Support for HTTPS and iSCSI . . . 11
Quick start for ONTAP Mediator . . . 11
What's next? . . . 12
Install and setup . . . 13
ONTAP Mediator installation workflow summary . . . 13
Prepare to install ONTAP Mediator . . . 13
Installation requirements . . . 14
Host requirements . . . 14
Firewall and networking requirements . . . 15
OS requirements . . . 16
Install ONTAP Mediator when UEFI Secure Boot is enabled . . . 17
Provide repository access for ONTAP Mediator installation . . . 17
Download the ONTAP Mediator installation package . . . 25
Verify the ONTAP Mediator code signature . . . 26
Install the ONTAP Mediator installation package . . . 28
Register the security key for UEFI Secure Boot . . . 49
Sign SCST kernel modules . . . 50
Verify the ONTAP Mediator installation status . . . 51
Use ONTAP Mediator . . . 53
Use ONTAP Mediator for SnapMirror active sync and MetroCluster IP . . . 53
Monitor SnapMirror active sync relationships . . . 53
Monitor MetroCluster IP configurations . . . 53
Administer ONTAP Mediator . . . 55
Upgrade ONTAP Mediator . . . 55
Upgrade ONTAP Mediator . . . 55
Upgrade the host OS and reinstall or upgrade ONTAP Mediator . . . 60
Configure ONTAP Mediator attributes . . . 65
ONTAP Mediator configurable settings . . . 65
Install third-party SSL certificates . . . 66
Configure password attack protection . . . 68
Configure password complexity rules . . . 69

Configure the free-space threshold 69
Configure reserved log space 69

ONTAP Mediator documentation

Release notes

What's new in ONTAP Mediator

New enhancements to ONTAP Mediator are provided with each release.

For OS and SCST version information, see the Support matrix.

ONTAP Mediator version	Enhancements
1.13	• Support for RHEL: ◦ Compatible: 10.1, 9.7, and 9.5. ◦ Recommended: 10.3, 10.2, 10.0, 9.9, 9.8, 9.6, 9.4, and 8.10. • Support for Rocky Linux 10.2, 10.1, 10.0, 9.8, 9.7, 9.6, 9.4, and 8.10. • Support for Oracle Linux 10.2 (rhck kernel), 10.1 (rhck and uek kernels), 10.0 (uek kernel), 9.8 (uek kernel), 9.7 (rhck and uek kernels), and 9.6 (rhck kernel). • The self-signed CA certificates now include a unique per-installation identifier in their common name, for example `ONTAP Mediator CA 1a2b3c4d`. This enables you to register several ONTAP Mediator instances with the same ONTAP cluster. If your ONTAP Mediator must pair with an ONTAP release that does not recognize the identifier, you can switch to the legacy fixed common name. Learn how to switch to the legacy CA certificate common name.
1.12.1	• Support for RHEL: ◦ Compatible: 10.1, 9.7, and 9.5. ◦ Recommended: 10.2, 10.0, 9.8, 9.6, 9.4, and 8.10. • Support for Rocky Linux 10.2, 10.1, 10.0, 9.8, and 8.10. • Support for Oracle Linux 10.1 and 9.8 with both rhck and uek kernels. • Beginning with ONTAP Mediator 1.12.1 and ONTAP 9.19.1, a single ONTAP Mediator instance configured with HTTPS can monitor the following configurations simultaneously: ◦ A maximum of ten four-node or five eight-node MetroCluster IP configurations if installed on a host with an 8-core CPU and at least 16GB of RAM. See Use ONTAP Mediator to monitor multiple MetroCluster IP configurations. ◦ One four-node MetroCluster IP configuration and one SnapMirror active sync peer relationship if installed on a host with a 2-core CPU. See Configure the ONTAP Mediator for SnapMirror active sync. • Includes a new tool to repair a broken SCST installation. After installing ONTAP Mediator, you can find the tool at `/usr/local/bin/mediator_repair_scst`. You can use the tool after an OS upgrade, if the upgraded OS requires an SCST patch that is not available in the default SCST package.

1.12	• Support for RHEL: ◦ Compatible: 10.1, 9.7, and 9.5. ◦ Recommended: 10.2, 10.0, 9.8, 9.6, 9.4, and 8.10. • Support for Rocky Linux 10.1, 10.0, 9.8, and 8.10. • Support for Oracle Linux 10.1 and 9.7 with both rhck and uek kernels. • Beginning with ONTAP 9.19.1 and ONTAP Mediator 1.12, you can configure an ONTAP Mediator that uses HTTPS for communication in MetroCluster IP configurations. HTTPS provides increased security by encrypting entire sessions using SSL/TLS. You can choose to install ONTAP Mediator with support for HTTPS only. The SCST package is not installed if this installation method is used. Learn more about HTTPS and iSCSI installation options for ONTAP Mediator.
1.11	• Support for RHEL: ◦ Compatible: 9.5. ◦ Recommended: 10.1, 10.0, 9.7, 9.6, 9.4, and 8.10. • Support for Rocky Linux 10.1, 10.0, 9.7, and 8.10. • Support for Oracle Linux 10.0 and 9.6 with both rhck and uek kernels. • Adds support for IPv6 for MetroCluster IP configurations. • Adds support for fapolicyd.
1.10	• Support for RHEL: ◦ Compatible: 9.5. ◦ Recommended: 10.0, 9.6, 9.4, and 8.10. • Support for Rocky Linux 10.0, 9.6, 9.4, and 8.10. • Upgrades the base Python version from Python 3.9 to Python 3.12.
1.9.1	• Support for RHEL: ◦ Compatible: 8.9, 8.7, 8.6, 8.5, and 8.4. ◦ Recommended: 9.5, 9.4, 9.2, 9.0, 8.10, and 8.8. • Support for Rocky Linux 9.5 and 8.10. • Adds new certificates for code signature verification. • Added support for skipping code signature checks using the `-skip-code-signature-check` flag. • The installer displays warnings when it detects expired code signature certificates.

1.9	• Support for RHEL: ◦ Compatible: 8.9, 8.7, 8.6, 8.5, and 8.4. ◦ Recommended: 9.4, 9.2, 9.0, 8.10, and 8.8. • Support for Rocky Linux 9.5 and 8.10. • FIPS support for RHEL and Rocky Linux. • Added performance enhancements for larger scalability. • Improved filenames to simplify the setup of PKI-signed certificates.
1.8	• Support for RHEL: ◦ Compatible: 8.7, 8.6, 8.5, and 8.4. ◦ Recommended: 9.4, 9.3, 9.2, 9.1, 9.0, 8.10, 8.9, and 8.8. • Support for Rocky Linux 9.4 and 8.10.
1.7	• Support for RHEL: ◦ Compatible: 8.7, 8.6, 8.5, and 8.4. ◦ Recommended: 9.3, 9.2, 9.1, 9.0, 8.9, and 8.8. • Support for Rocky Linux 9.3 and 8.9. • Support for SAN (Subject Alternative Name) data in self-signed certificates and third-party signed certificates.
1.6	• Python 3.9 updates. • Support for RHEL: ◦ Compatible: 8.7, 8.6, 8.5, and 8.4. ◦ Recommended: 9.2, 9.1, 9.0, and 8.8. • Support for Rocky Linux 9.2 and 8.8. • Discontinued support for RHEL 7.x / CentOS all releases.
1.5	• Support for RHEL 8.5, 8.4, 8.3, 8.2, 8.1, 8.0, 7.9, 7.8, 7.7, and 7.6. • Support for CentOS 7.9, 7.8, 7.7, and 7.6. • Includes deprecation warnings for RHEL 7.x / CentOS 7.x. • Optimizes speed for larger scale SnapMirror active sync systems. • A cryptographic code-signature was added to the installer.
1.4	• Support for RHEL 8.5, 8.4, 8.3, 8.2, 8.1, 8.0, 7.9, 7.8, 7.7, and 7.6. • Support for CentOS 7.9, 7.8, 7.7, and 7.6. • Added support for UFEI-based firmware’s Secure Boot (SB).
1.3	• Support for RHEL 8.3, 8.2, 8.1, 8.0, 7.9, 7.8, 7.7, and 7.6. • Support for CentOS 7.9, 7.8, 7.7, and 7.6.
1.2	• Support for RHEL 8.1, 8.0, 7.9, 7.8, 7.7, and 7.6. • Support for CentOS 7.9, 7.8, 7.7, and 7.6. • Support for HTTPs mailboxes. • For use with ONTAP 9.8+ MCC-IP AUSO and SnapMirror active sync ZRTO.

1.1	• Support for RHEL 8.0 and 7.6. • Support for CentOS 7.6. • Eliminates Perl dependencies.
1.0	• Support for iSCSI mailboxes. • For use with ONTAP 9.7+ MCC-IP AUSO. • Support for RHEL/CentOS 7.6.

ONTAP Mediator support matrix

The ONTAP Mediator support matrix includes operating system and SCST version support information.

OS support matrix

The following table lists the supported operating systems for each version of ONTAP Mediator.

OS for ONTAP Mediator	**1.13**	**1.12.1**	**1.12**	**1.11**	**1.10**	**1.9.1**	**1.9**	**1.8**	**1.7**	**1.6**	**1.5**	**1.4**	**1.3**	**1.2**	**1.1**	**1.0**
RHEL 10.3	Yes	No	No	No	No	No	No	No	No	No	No	No	No	No	No	No
RHEL 10.2	Yes	Yes	Yes	No	No	No	No	No	No	No	No	No	No	No	No	No
RHEL 10.1	Compatible	Compatible	Compatible	Yes	No	No	No	No	No	No	No	No	No	No	No	No
RHEL 10.0	Yes	Yes	Yes	Yes	Yes	No	No	No	No	No	No	No	No	No	No	No
RHEL 9.9	Yes	No	No	No	No	No	No	No	No	No	No	No	No	No	No	No
RHEL 9.8	Yes	Yes	Yes	No	No	No	No	No	No	No	No	No	No	No	No	No
RHEL 9.7	Compatible	Compatible	Compatible	Yes	No	No	No	No	No	No	No	No	No	No	No	No
RHEL 9.6	Yes	Yes	Yes	Yes	Yes	No	No	No	No	No	No	No	No	No	No	No

RHEL 9.5	Compatible	Compatible	Compatible	Compatible	Compatible	Yes	No	No	No	No	No	No	No	No	No	No
RHEL 9.4	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	No	No	No	No	No	No	No	No
RHEL 9.3	No	No	No	No	No	No	No	Yes	Yes	No	No	No	No	No	No	No
RHEL 9.2	No	No	No	No	No	Yes	Yes	Yes	Yes	Yes	No	No	No	No	No	No
RHEL 9.1	No	No	No	No	No	No	No	Yes	Yes	Yes	No	No	No	No	No	No
RHEL 9.0	No	No	No	No	No	Yes	Yes	Yes	Yes	Yes	No	No	No	No	No	No
RHEL 8.10	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	No	No	No	No	No	No	No	No
RHEL 8.9	No	No	No	No	No	Compatible	Compatible	Yes	Yes	No	No	No	No	No	No	No
RHEL 8.8	No	No	No	No	No	Yes	Yes	Yes	Yes	Yes	No	No	No	No	No	No
RHEL 8.7	No	No	No	No	No	Compatible	Compatible	Compatible	Compatible	Compatible	No	No	No	No	No	No
RHEL 8.6	No	No	No	No	No	Compatible	Compatible	Compatible	Compatible	Compatible	No	No	No	No	No	No
RHEL 8.5	No	No	No	No	No	Compatible	Compatible	Compatible	Compatible	Compatible	Yes	Yes	No	No	No	No
RHEL 8.4	No	No	No	No	No	Compatible	Compatible	Compatible	Compatible	Compatible	Yes	Yes	No	No	No	No
RHEL 8.3	Obsolete	Obsolete	Obsolete	Obsolete	Obsolete	Obsolete	Obsolete	Obsolete	Obsolete	Obsolete	Yes	Yes	Yes	No	No	No
RHEL 8.2	Obsolete	Obsolete	Obsolete	Obsolete	Obsolete	Obsolete	Obsolete	Obsolete	Obsolete	Obsolete	Yes	Yes	Yes	No	No	No
RHEL 8.1	Obsolete	Obsolete	Obsolete	Obsolete	Obsolete	Obsolete	Obsolete	Obsolete	Obsolete	Obsolete	Yes	Yes	Yes	Yes	No	No
RHEL 8.0	Obsolete	Obsolete	Obsolete	Obsolete	Obsolete	Obsolete	Obsolete	Obsolete	Obsolete	Obsolete	Yes	Yes	Yes	Yes	Yes	No

RHEL and CentOS 7.9	Obsolete	Obsolete	Obsolete	Obsolete	Obsolete	Obsolete	Obsolete	Obsolete	Obsolete	Obsolete	Yes	Yes	Yes	Compatible	No	No
RHEL and CentOS 7.8	Obsolete	Obsolete	Obsolete	Obsolete	Obsolete	Obsolete	Obsolete	Obsolete	Obsolete	Obsolete	Yes	Yes	Yes	Yes	No	No
RHEL and CentOS 7.7	Obsolete	Obsolete	Obsolete	Obsolete	Obsolete	Obsolete	Obsolete	Obsolete	Obsolete	Obsolete	Yes	Yes	Yes	Yes	No	No
RHEL and CentOS 7.6	Obsolete	Obsolete	Obsolete	Obsolete	Obsolete	Obsolete	Obsolete	Obsolete	Obsolete	Obsolete	Yes	Yes	Yes	Yes	Yes	Yes (RHEL only)
CentOS 8 and stream	No	No	No	No	No	No	No	No	No	No	No	No	No	N/A	N/A	N/A
Rocky Linux 10	Yes (10.0, 10.1, and 10.2)	Yes (10.0, 10.1, and 10.2)	Yes (10.0 and 10.1)	Yes (10.0 and 10.1)	Yes (10.0)	No	No	No	No	No	No	No	No	No	No	No
Rocky Linux 9	Yes (9.4, 9.6, 9.7, and 9.8)	Yes (9.8)	Yes (9.8)	Yes (9.7)	Yes (9.6 and 9.4)	Yes (9.5)	Yes (9.5)	Yes (9.4)	Yes (9.3)	Yes (9.2)	N/A	N/A	N/A	N/A	N/A	N/A
Rocky Linux 8	Yes (8.10)	Yes (8.10)	Yes (8.10)	Yes (8.10)	Yes (8.10)	Yes (8.10)	Yes (8.10)	Yes (8.10)	Yes (8.9)	Yes (8.8)	N/A	N/A	N/A	N/A	N/A	N/A

Oracle Linux 10	Yes (10.0, 10.1, and 10.2)	Yes (10.1)	Yes (10.1)	Yes (10.0)	No	No	No	No	No	No	No	No	No	No	No	No
Oracle Linux 9	Yes (9.6, 9.7, and 9.8)	Yes (9.8)	Yes (9.7)	Yes (9.6)	No	No	No	No	No	No	No	No	No	No	No	No

- "Yes" means that the OS is recommended for ONTAP Mediator installation and is fully compatible and supported.
- "No" means that the OS and ONTAP Mediator are not compatible.
- "Compatible" means that Red Hat no longer supports these RHEL versions, but ONTAP Mediator can still be installed on them.
- ONTAP Mediator 1.6 adds support for Rocky Linux 9 and 8.
- ONTAP Mediator 1.5 was the last supported release for RHEL 7.x branch operating systems.
- Centos 8 was removed for all releases due to its rebranching. Centos Stream was deemed as not a suitable production target OS. No support is planned.

SCST support matrix

The following table shows the supported SCST version for each version of ONTAP Mediator.

ONTAP Mediator version	Supported SCST version
ONTAP Mediator 1.13	scst-3.10.tar.gz
ONTAP Mediator 1.12.1	scst-3.10.tar.gz
ONTAP Mediator 1.12	scst-3.10.tar.gz
ONTAP Mediator 1.11	scst-3.9.tar.gz
ONTAP Mediator 1.10	scst-3.9.tar.gz
ONTAP Mediator 1.9.1	scst-3.8.0.tar.bz2
ONTAP Mediator 1.9	scst-3.8.0.tar.bz2
ONTAP Mediator 1.8	scst-3.8.0.tar.bz2
ONTAP Mediator 1.7	scst-3.7.0.tar.bz2
ONTAP Mediator 1.6	scst-3.7.0.tar.bz2
ONTAP Mediator 1.5	scst-3.6.0.tar.bz2
ONTAP Mediator 1.4	scst-3.6.0.tar.bz2
ONTAP Mediator 1.3	scst-3.5.0.tar.bz2
ONTAP Mediator 1.2	scst-3.4.0.tar.bz2
ONTAP Mediator 1.1	scst-3.4.0.tar.bz2

ONTAP Mediator version	Supported SCST version
ONTAP Mediator 1.0	scst-3.3.0.tar.bz2

Get started

Learn about ONTAP Mediator

ONTAP Mediator supports ONTAP features by providing a persistent and fenced store for high-availability (HA) metadata, synchronous node health query functionality to aid in quorum determination, and serving as a ping proxy for controller liveliness.

This documentation refers to the on-premise version of ONTAP Mediator. For information about ONTAP Cloud Mediator, available beginning with ONTAP 9.17.1, see the SnapMirror active sync documentation.

How ONTAP Mediator works

ONTAP Mediator runs on a Linux host in a third failure domain, separate from your ONTAP clusters, where it provides a persistent, fenced store for HA metadata. ONTAP uses this store, along with the Mediator's node health query and ping-proxy functions, to monitor SnapMirror active sync relationships and MetroCluster IP configurations.

- ONTAP Mediator uses the node management LIF and cluster management LIF to communicate with each cluster over multiple, redundant paths. This lets ONTAP distinguish between a full site failure and a single InterSwitch Link (ISL) failure.
- In SnapMirror active sync, if a cluster loses connection to both its peer and ONTAP Mediator, ONTAP treats the peer as unreachable and can automatically fail over to the surviving copy.
- In MetroCluster IP configurations, ONTAP Mediator assists with a Mediator-assisted automatic unplanned switchover (MAUSO) by storing state information about the MetroCluster nodes and helping determine the appropriate course of action during a failure.

ONTAP Mediator services

ONTAP Mediator provides two additional systemctl services:

- **`ontap_mediator.service`**

  Maintains the REST API server for managing the ONTAP relationships.

- **`mediator-scst.service`**

  Controls the startup and shutdown of the iSCSI module (SCST).

Tools provided for the system administrator

Tools provided for the system administrator:

- **`/usr/local/bin/mediator_change_password`**

  Sets a new API password when the current API username and password are provided.

- **`/usr/local/bin/mediator_change_user`**

Sets a new API username when the current API username and password are provided.

- **`/usr/local/bin/mediator_generate_support_bundle`**

  Generates a local tgz file containing all useful support information needed for communication with NetApp customer support. This includes application configuration, logs, and some system information. The bundles are generated on the local disk and can be transferred manually, as needed. Storage location: /opt/netapp/data/support_bundles/

- **`/usr/local/bin/uninstall_ontap_mediator`**

  Removes the ONTAP Mediator package and the SCST kernel module. This includes all configuration, logs, and mailbox data.

- **`/usr/local/bin/mediator_unlock_user`**

  Releases a lock-out on the API user account if the authentication retry limit was reached. This feature is used to prevent brute force password derivation. It prompts the user for the correct username and password.

- **`/usr/local/bin/mediator_add_user`**

  (Support only) Used to add the API user upon installation.

SCST and kernel updates

ONTAP Mediator uses SCST to provide iSCSI for communication in MetroCluster IP configurations. The SCST package is a kernel module that is compiled during installation.
You might need to reinstall SCST after updates to the kernel. Alternatively, uninstall ONTAP Mediator, then re-install and reconfigure the ONTAP relationship. Refer to SCST: A Generic SCSI Target Subsystem for Linux to learn more about SCST.

> You should coordinate any updates to the server OS kernel with a maintenance window in ONTAP.

Support for HTTPS and iSCSI

Beginning with ONTAP 9.19.1 and ONTAP Mediator 1.12, you can configure an ONTAP Mediator that uses HTTPS or iSCSI to monitor a MetroCluster IP configuration. You can choose to install ONTAP Mediator with support for both iSCSI and HTTPS or HTTPS only. If ONTAP Mediator is installed with support for HTTPS only, the SCST package is not installed. Refer to HTTPS and iSCSI installation options for ONTAP Mediator for more information.

Quick start for ONTAP Mediator

To get started with ONTAP Mediator, you install the software and configure it to work with your ONTAP environment.

1 **Install ONTAP Mediator**

Install ONTAP Mediator on your system and ensure that all required dependencies are met.

2 **Verify the ONTAP Mediator installation**

Verify that the ONTAP Mediator installation was successful by checking the service status and reviewing the logs for any errors or warnings.

3 **Configure ONTAP Mediator for your ONTAP environment**

Depending on whether you want to use ONTAP Mediator for MetroCluster IP, SnapMirror active sync, or both, follow the appropriate configuration steps to ensure that all necessary settings are correctly applied.

What's next?

You can now use ONTAP Mediator to monitor MetroCluster IP configurations, SnapMirror active sync configurations, or both. For more information, refer to the SnapMirror active sync documentation and MetroCluster documentation.

Install and setup

ONTAP Mediator installation workflow summary

Installing ONTAP Mediator includes preparing for the installation, providing access to repositories, downloading the installation package, verifying the code signature, and installing the ONTAP Mediator package.

1. **Prepare to install ONTAP Mediator**

To install ONTAP Mediator, you must ensure that all prerequisites are met.

2. **Provide repository access**

You should enable access to repositories so that ONTAP Mediator can access the required packages during the installation process.

3. **Download the ONTAP Mediator installation package**

Download the ONTAP Mediator installation package from the ONTAP Mediator download page.

4. **Verify the code signature of the ONTAP Mediator installation package**

NetApp recommends verifying the ONTAP Mediator code signature before installing the ONTAP Mediator installation package.

5. **Install the ONTAP Mediator installation package**

Download the installation package and run the installer on the host.

6. **Verify the ONTAP Mediator installation**

After you install ONTAP Mediator, verify that it is running successfully.

Prepare to install ONTAP Mediator

To install ONTAP Mediator, you need to meet all prerequisites, download the installation package, and run the installer on the host.

- To upgrade ONTAP Mediator to a new version, see Upgrade ONTAP Mediator.
- If you need to upgrade the host OS and reinstall or upgrade ONTAP Mediator as part of that process, see Upgrade the host OS and ONTAP Mediator.

Installation requirements

Review the following requirements before installing ONTAP Mediator.

- ONTAP Mediator 1.8 and earlier is not compatible with Red Hat Enterprise Linux (RHEL) FIPS mode and prevents it from installing successfully. You can check if FIPS mode is enabled using the `fips-mode-setup --check` command. You can disable FIPS mode using the `fips-mode-setup --disable` command. Reboot after disabling FIPS mode to successfully install ONTAP Mediator 1.8 or earlier.
- If the installation fails, you might need to try a newer version of ONTAP Mediator.
- From June 15, 2025, you can't install or upgrade ONTAP Mediator 1.9 and 1.8 because their code signing certificates have expired. If the installation fails, use the ONTAP Mediator 1.9.1 patch version instead.

- If you install the `yum-utils` package, you can use the `needs-restarting` command.
- Beginning with ONTAP Mediator 1.11, IPv6 is supported for MetroCluster IP configurations.

HTTPS and iSCSI installation options for ONTAP Mediator

Before installing ONTAP Mediator, review the following information:

- In ONTAP Mediator, the SCST package is used to provide iSCSI for communication in MetroCluster IP configurations. The SCST package is a kernel module that is compiled during installation. Beginning with ONTAP 9.19.1 and Mediator 1.12, you can configure an ONTAP Mediator that uses HTTPS instead of iSCSI to monitor a MetroCluster IP configuration.
- Beginning with ONTAP 9.19.1 and ONTAP Mediator 1.12, you have the option to install ONTAP Mediator with iSCSI and HTTPS support, or support for HTTPS only. If you choose to install ONTAP Mediator with support for HTTPS only, the SCST package is not installed.
- If you upgrade from ONTAP Mediator 1.11 or earlier with SCST installed, and you choose the HTTPS only option during the upgrade, the SCST package is uninstalled and the new ONTAP Mediator version does not contain the SCST package.
- If you plan to use ONTAP Mediator to monitor multiple MetroCluster IP configurations, the support depends on your Mediator type, ONTAP version, and OS. Refer to Use ONTAP Mediator to monitor multiple MetroCluster IP configurations for more information.
- Installing ONTAP Mediator with support for HTTPS only is supported for the following use cases:
  - You plan to only use ONTAP Mediator to monitor your SnapMirror active sync clusters.
  - You plan to configure ONTAP Mediator with HTTPS to monitor one or more MetroCluster IP configurations.
- ONTAP Mediator must be installed with support for iSCSI and HTTPS in the following scenario:
  - You plan to configure ONTAP Mediator with iSCSI to monitor one or more MetroCluster IP configurations.

Host requirements

Follow these requirements when installing RHEL or Rocky Linux and configuring the associated repositories.

If you modify the installation or configuration process, you might need to perform additional steps.

Linux distribution requirements

- Install RHEL or Rocky Linux according to Red Hat's best practices. Because CentOS 8.x has reached end-of-life, compatible versions of CentOS 8.x are not recommended.
- When installing ONTAP Mediator, ensure the system has access to the required repository so the installation program can retrieve and install all required software dependencies.
- To enable the yum installer to find dependent software in the RHEL repositories, register the system during installation or afterwards using a valid Red Hat subscription.

See the Red Hat Subscription Manager documentation for further information.

Firewall and networking requirements

ONTAP Mediator uses a number of ports to communicate with specific services.

- If you are using a third-party firewall:
  - HTTPS access must be enabled.
  - It must be configured to allow access on ports 31784 and 3260.

    When using the default Red Hat or CentOS firewall, the firewall is automatically configured during Mediator installation.
- For Linux hosts without internet access, make sure the required packages are available in a local repository.

  If you are using Link Aggregation Control Protocol (LACP) in a Linux environment, configure the kernel and set the `sysctl net.ipv4.conf.all.arp_ignore` to `2`.

The following table lists the ports that you must allow in your firewall:

- The iSCSI port is only required in a MetroCluster IP configuration.
- The 22/tcp port is not required for normal operation but you can enable it temporarily for maintenance and disable it when the maintenance session has finished.

Port/services	Source	Direction	Destination	Purpose
22/tcp	Management host	Inbound	ONTAP Mediator	(Optional) SSH / ONTAP Mediator management
31784/tcp	cluster-mgmt and node-mgmt LIFs	Inbound	ONTAP Mediator web server	(Required) REST API (HTTPS)
3260/tcp	node-mgmt LIFs	Inbound	ONTAP Mediator iSCSI targets	(Required for MetroCluster IP configurations) iSCSI data connection for mailboxes

For a SnapMirror active sync or MetroCluster IP ONTAP Mediator that uses HTTPS for communication, ONTAP doesn't require port 3260 to be enabled or connected.

OS requirements

Your OS must meet the following requirements:

- 64-bit physical installation or virtual machine
- 8 GB RAM
- 1 GB disk space (used for applications installation, server logs, and the database)
- User: Root access

Review the OS support matrix for supported OSs for each version of ONTAP Mediator.

OS required packages

The following packages are required by ONTAP Mediator:

The packages are either pre-installed or automatically installed by the ONTAP Mediator installer.

All RHEL/CentOS versions	Additional packages for RHEL 10.x / Rocky Linux 10	Additional packages for RHEL 9.x / Rocky Linux 9	Additional packages for RHEL 8.x / Rocky Linux 8
• openssl • openssl-devel • kernel-devel-$ (uname -r) • gcc • make • libselinux-utils • patch • bzip2 • perl-Data-Dumper • perl-ExtUtils-MakeMaker • efibootmgr • mokutil	• python3.12 • python3.12-devel	• elfutils-libelf-devel • policycoreutils-python-utils • python3 • python3-devel	• elfutils-libelf-devel • policycoreutils-python-utils • redhat-lsb-core • python39 • python39-devel

The Mediator installation package is a self-extracting compressed tar file that includes:

- An RPM file containing all dependencies that cannot be obtained from the supported release's repository.
- An install script.

A valid SSL certificate is recommended.

- The kernel version must match the operating system version.
- Do not upgrade the kernel past the supported OS version for your ONTAP Mediator release as the tested SCST module might not work.

Install ONTAP Mediator when UEFI Secure Boot is enabled

ONTAP Mediator can be installed on a system with or without UEFI Secure Boot enabled.

About this task

- You can choose to disable UEFI Secure Boot before installing ONTAP Mediator if it is not needed or if you are troubleshooting ONTAP Mediator installation issues. Disable the UEFI Secure Boot option from your machine settings.
- If you installed ONTAP Mediator 1.12 or later with support for HTTPS only, the SCST package is not installed and you can skip this task.

For detailed instructions on disabling UEFI Secure Boot, refer to the documentation for your host OS.

To install ONTAP Mediator with UEFI Secure Boot enabled, you must register a security key before the service can start. The key is generated during the SCST installation's compile step and saved as a private-public key pair on your machine. Use the `mokutil` utility to add the public key as a Machine Owner Key (MOK) to your UEFI firmware, enabling the system to trust and load the signed module. Save the `mokutil` passphrase in a secure location as this is required when rebooting your system to activate the MOK.

Steps

1. Check if UEFI Secure Boot is enabled on your system:

```
mokutil --sb-state
```

The results indicate whether UEFI Secure Boot is enabled on this system.

If…	Go to…
UEFI secure boot is enabled	The step where you run the `mokutil` utility
UEFI secure boot is disabled	Provide repository access for ONTAP Mediator installation

- You are prompted to create a passphrase that you must store in a secure location. You'll need this passphrase to enable the key in the UEFI Boot Manager.
- ONTAP Mediator 1.2.0 and earlier versions do not support this mode.

2. If the `mokutil` utility is not installed, run the following command:

```
yum install mokutil
```

What's next?

Provide repository access for ONTAP Mediator installation

Provide repository access for ONTAP Mediator installation

You should enable access to repositories so that ONTAP Mediator can access the

required packages during the installation process.

Steps

1. Determine which repositories must be accessed, as shown in the following table:

If your operating system is…	You must provide access to these repositories…
Oracle Linux 10.x	• ol-10-for-x86_64-baseos-rpms • ol-10-for-x86_64-appstream-rpms
Oracle Linux 9.x	• ol-9-for-x86_64-baseos-rpms • ol-9-for-x86_64-appstream-rpms
RHEL 10.x	• rhel-10-for-x86_64-baseos-rpms • rhel-10-for-x86_64-appstream-rpms
RHEL 9.x	• rhel-9-for-x86_64-baseos-rpms • rhel-9-for-x86_64-appstream-rpms
RHEL 8.x	• rhel-8-for-x86_64-baseos-rpms • rhel-8-for-x86_64-appstream-rpms
RHEL 7.x	• rhel-7-server-optional-rpms
CentOS 7.x	• C7.6.1810 - Base repository
Rocky Linux 10	• appstream • baseos
Rocky Linux 9	• appstream • baseos
Rocky Linux 8	• appstream • baseos

2. Use one of the following procedures to enable access to the repositories listed above so ONTAP Mediator can access the required packages during the installation process.

> If ONTAP Mediator has dependencies on Python modules present in the "extras" and "optional" repositories, it might need to access the `rhel-X-for-x86_64-extras-rpms` and `rhel-X-for-x86_64-optional-rpms` files.

▼ **Procedure for Oracle Linux 10.x operating system**

Use this procedure if your operating system is **Oracle Linux 10.x** to enable access to repositories:

Steps

1. Subscribe to the required repository:

```
subscription-manager repos --enable ol-10-for-x86_64-baseos-rpms
```

```
subscription-manager repos --enable ol-10-for-x86_64-appstream-rpms
```

The following example shows the execution of this command:

```
[root@localhost ~]# subscription-manager repos --enable ol-10-for-
x86_64-baseos-rpms
Repository 'ol-10-for-x86_64-baseos-rpms' is enabled for this
system.
[root@localhost ~]# subscription-manager repos --enable ol-10-for-
x86_64-appstream-rpms
Repository 'ol-10-for-x86_64-appstream-rpms' is enabled for this
system.
```

2. Run the following command to list the enabled repositories:

```
yum repolist
```

The newly subscribed repositories should appear in the list.

▼ **Procedure for Oracle Linux 9.x operating system**

Use this procedure if your operating system is **Oracle Linux 9.x** to enable access to repositories:

Steps

1. Subscribe to the required repository:

```
subscription-manager repos --enable ol-9-for-x86_64-baseos-rpms
```

```
subscription-manager repos --enable ol-9-for-x86_64-appstream-rpms
```

The following example shows the execution of this command:

```
[root@localhost ~]# subscription-manager repos --enable ol-9-for-
x86_64-baseos-rpms
Repository 'ol-9-for-x86_64-baseos-rpms' is enabled for this
system.
[root@localhost ~]# subscription-manager repos --enable ol-9-for-
x86_64-appstream-rpms
Repository 'ol-9-for-x86_64-appstream-rpms' is enabled for this
system.
```

2. Run the following command to list the enabled repositories:

```
yum repolist
```

The newly subscribed repositories should appear in the list.

▼ **Procedure for RHEL 10.x operating system**

Use this procedure if your operating system is **RHEL 10.x** to enable access to repositories:

Steps

1. Subscribe to the required repository:

```
subscription-manager repos --enable rhel-10-for-x86_64-baseos-rpms
```

```
subscription-manager repos --enable rhel-10-for-x86_64-appstream-
rpms
```

The following example shows the execution of this command:

```
[root@localhost ~]# subscription-manager repos --enable rhel-10-
for-x86_64-baseos-rpms
Repository 'rhel-10-for-x86_64-baseos-rpms' is enabled for this
system.
[root@localhost ~]# subscription-manager repos --enable rhel-10-
for-x86_64-appstream-rpms
Repository 'rhel-10-for-x86_64-appstream-rpms' is enabled for this
system.
```

2. Run the following command to list the enabled repositories:

```
yum repolist
```

The newly subscribed repositories should appear in the list.

▼ **Procedure for RHEL 9.x operating system**

Use this procedure if your operating system is **RHEL 9.x** to enable access to repositories:

Steps

1. Subscribe to the required repository:

```
subscription-manager repos --enable rhel-9-for-x86_64-baseos-rpms
```

```
subscription-manager repos --enable rhel-9-for-x86_64-appstream-
```

```
rpms
```

The following example shows the execution of this command:

```
[root@localhost ~]# subscription-manager repos --enable rhel-9-for-
x86_64-baseos-rpms
Repository 'rhel-9-for-x86_64-baseos-rpms' is enabled for this
system.
[root@localhost ~]# subscription-manager repos --enable rhel-9-for-
x86_64-appstream-rpms
Repository 'rhel-9-for-x86_64-appstream-rpms' is enabled for this
system.
```

2. Run the following command to list the enabled repositories:

```
yum repolist
```

The newly subscribed repositories should appear in the list.

▼ **Procedure for RHEL 8.x operating system**

Use this procedure if your operating system is **RHEL 8.x** to enable access to repositories:

Steps

1. Subscribe to the required repository:

```
subscription-manager repos --enable rhel-8-for-x86_64-baseos-rpms
```

```
subscription-manager repos --enable rhel-8-for-x86_64-appstream-
rpms
```

The following example shows the execution of this command:

```
[root@localhost ~]# subscription-manager repos --enable rhel-8-for-
x86_64-baseos-rpms
Repository 'rhel-8-for-x86_64-baseos-rpms' is enabled for this
system.
[root@localhost ~]# subscription-manager repos --enable rhel-8-for-
x86_64-appstream-rpms
Repository 'rhel-8-for-x86_64-appstream-rpms' is enabled for this
system.
```

2. Run the following command to list the enabled repositories:

```
yum repolist
```

The newly subscribed repositories should appear in the list.

▼ **Procedure for RHEL 7.x operating system**

Use this procedure if your operating system is **RHEL 7.x** to enable access to repositories:

Steps

1. Subscribe to the required repository:

```
subscription-manager repos --enable rhel-7-server-optional-rpms
```

The following example shows the execution of this command:

```
[root@localhost ~]# subscription-manager repos --enable rhel-7-
server-optional-rpms
Repository 'rhel-7-server-optional-rpms' is enabled for this
system.
```

2. Run the following command to list the enabled repositories:

```
yum repolist
```

The following example shows the execution of this command. The "rhel-7-server-optional-rpms" repository should appear in the list.

```
[root@localhost ~]# yum repolist
Loaded plugins: product-id, search-disabled-repos, subscription-
manager
rhel-7-server-optional-rpms | 3.2 kB  00:00:00
rhel-7-server-rpms | 3.5 kB  00:00:00
(1/3): rhel-7-server-optional-rpms/7Server/x86_64/group             |
26 kB  00:00:00
(2/3): rhel-7-server-optional-rpms/7Server/x86_64/updateinfo        |
2.5 MB  00:00:00
(3/3): rhel-7-server-optional-rpms/7Server/x86_64/primary_db        |
8.3 MB  00:00:01
repo id                                    repo name
status
rhel-7-server-optional-rpms/7Server/x86_64   Red Hat Enterprise Linux
7 Server - Optional (RPMs)   19,447
```

```
rhel-7-server-rpms/7Server/x86_64            Red Hat Enterprise Linux
7 Server (RPMs)              26,758
repolist: 46,205
[root@localhost ~]#
```

▼ **Procedure for CentOS 7.x operating system**

Use this procedure if your operating system is **CentOS 7.x** to enable access to repositories:

> The following examples are showing a repository for CentOS 7.6 and might not work for other CentOS versions. Use the base repository for your version of CentOS.

Steps

1. Add the C7.6.1810 - Base repository. The C7.6.1810 - Base vault repository contains the "kernel-devel" package needed for ONTAP Mediator.
2. Add the following lines to /etc/yum.repos.d/CentOS-Vault.repo.

```
[C7.6.1810-base]
name=CentOS-7.6.1810 - Base
baseurl=http://vault.centos.org/7.6.1810/os/$basearch/
gpgcheck=1
gpgkey=file:///etc/pki/rpm-gpg/RPM-GPG-KEY-CentOS-7
enabled=1
```

3. Run the following command to list the enabled repositories:

```
yum repolist
```

The following example shows the execution of this command. The CentOS-7.6.1810 - Base repository should appear in the list.

```
Loaded plugins: fastestmirror
Loading mirror speeds from cached hostfile
 * base: distro.ibiblio.org
 * extras: distro.ibiblio.org
 * updates: ewr.edge.kernel.org
C7.6.1810-base                                | 3.6 kB  00:00:00
(1/2): C7.6.1810-base/x86_64/group_gz         | 166 kB  00:00:00
(2/2): C7.6.1810-base/x86_64/primary_db       | 6.0 MB  00:00:04
repo id                     repo name                 status
C7.6.1810-base/x86_64       CentOS-7.6.1810 - Base  10,019
base/7/x86_64               CentOS-7 - Base         10,097
extras/7/x86_64             CentOS-7 - Extras       307
updates/7/x86_64            CentOS-7 - Updates      1,010
repolist: 21,433
```

```
[root@localhost ~]#
```

▼ **Procedure for Rocky Linux 10, 9, or 8 operating systems**

Use this procedure if your operating system is **Rocky Linux 10**, **Rocky Linux 9**, or **Rocky Linux 8** to enable access to repositories:

Steps

1. Subscribe to the required repositories:

```
dnf config-manager --set-enabled baseos
```

```
dnf config-manager --set-enabled appstream
```

2. Perform a `clean` operation:

```
dnf clean all
```

3. Verify the list of repositories:

```
dnf repolist
```

Example for Rocky Linux 10

```
[root@localhost ~]# dnf config-manager --set-enabled baseos
[root@localhost ~]# dnf config-manager --set-enabled appstream
[root@localhost ~]# dnf clean all
[root@localhost ~]# dnf repolist
repo id                         repo name
appstream                       Rocky Linux 10 - AppStream
baseos                          Rocky Linux 10 - BaseOS
[root@localhost ~]#
```

Example for Rocky Linux 9

```
[root@localhost ~]# dnf config-manager --set-enabled baseos
[root@localhost ~]# dnf config-manager --set-enabled appstream
[root@localhost ~]# dnf clean all
[root@localhost ~]# dnf repolist
repo id                         repo name
appstream                       Rocky Linux 9 - AppStream
baseos                          Rocky Linux 9 - BaseOS
```

```
[root@localhost ~]#
```

Example for Rocky Linux 8

```
[root@localhost ~]# dnf config-manager --set-enabled baseos
[root@localhost ~]# dnf config-manager --set-enabled appstream
[root@localhost ~]# dnf clean all
[root@localhost ~]# dnf repolist
repo id                         repo name
appstream                       Rocky Linux 8 - AppStream
baseos                          Rocky Linux 8 - BaseOS
[root@localhost ~]#
```

What's next?

Download the ONTAP Mediator installation package

Download the ONTAP Mediator installation package

Download the ONTAP Mediator installation package and install it.

Steps

1. Download the ONTAP Mediator installation package from the ONTAP Mediator download page.

   ONTAP Mediator download page

2. Place the Mediator installation package in the current working directory:

   ```
   ls ontap-mediator-1.13.0.tgz
   ```

   ```
   ontap-mediator-1.13.0.tgz
   ```

   For ONTAP Mediator versions 1.4 and earlier, the installer is named `ontap-mediator`.

   If your system does not have internet access, make sure the installer can access the required packages.

3. Move the Mediator installation package to the installation directory if needed.
4. Unzip the installer package:

   ```
   tar xvfz ontap-mediator-1.13.0.tgz
   ```

   ```
   ontap-mediator-1.13.0/
   ontap-mediator-1.13.0/csc-prod-ONTAP-Mediator.pem
   ```

```
ontap-mediator-1.13.0/csc-prod-chain-ONTAP-Mediator.pem
ontap-mediator-1.13.0/tsa-prod-ONTAP-Mediator.pem
ontap-mediator-1.13.0/tsa-prod-chain-ONTAP-Mediator.pem
ontap-mediator-1.13.0/ONTAP-Mediator-production.pub
ontap-mediator-1.13.0/ontap-mediator-1.13.0
ontap-mediator-1.13.0/ontap-mediator-1.13.0.sig.tsr
ontap-mediator-1.13.0/ontap-mediator-1.13.0.tsr
ontap-mediator-1.13.0/ontap-mediator-1.13.0.sig
```

What's next?

Verify the ONTAP Mediator code signature

Verify the ONTAP Mediator code signature

NetApp recommends verifying the ONTAP Mediator code signature before installation. This step is optional.

Before you begin

Make sure your system meets these needs before verifying the ONTAP Mediator code signature.

- From June 15th, 2025, you can't install or upgrade to ONTAP Mediator 1.9 and 1.8 because the code signature verification certificates have expired. Instead, install or upgrade to ONTAP Mediator 1.10 or later.
- If the system doesn't meet the below requirements, the verification process isn't required and you can go directly to Install the ONTAP Mediator installation package.

- openssl versions 1.0.2 to 3.0 for basic verification
- openssl version 1.1.0 or later for Time Stamping Authority (TSA) operations
- Public internet access for OCSP verification

The download package includes the following files:

File	Description
`ONTAP-Mediator-production.pub`	The public key used to verify the signature
`csc-prod-chain-ONTAP-Mediator.pem`	The public certification CA chain of trust
`csc-prod-ONTAP-Mediator.pem`	The certificate used to generate the key
`ontap-mediator-1.13.0`	The product installation executable for version 1.13
`ontap-mediator-1.13.0.sig`	The SHA-256 hashed, then RSA-signed using the csc-prod key, signature for the installer
`ontap-mediator-1.13.0.sig.tsr`	The revocation request for use by OCSCP for the installer's signature
`ontap-mediator-1.13.0.tsr`	The timestamp signing request file
`tsa-prod-ONTAP-Mediator.pem`	The public certificate for the TSR

`tsa-prod-chain-ONTAP-Mediator.pem`	The public certificate CA Chain for the TSR

Steps

1. Perform the revocation check on `csc-prod-ONTAP-Mediator.pem` by using Online Certificate Status Protocol (OCSP).
   a. Find the OCSP URL for the certificate. Developer certificates might not provide a URI:

   ```
   openssl x509 -noout -ocsp_uri -in csc-prod-chain-ONTAP-Mediator.pem
   ```

   b. Generate an OCSP request for the certificate.

   ```
   openssl ocsp -issuer csc-prod-chain-ONTAP-Mediator.pem -CAfile csc-
   prod-chain-ONTAP-Mediator.pem -cert csc-prod-ONTAP-Mediator.pem
   -reqout req.der
   ```

   c. Connect to the OCSP Manager to send the OCSP request:

   ```
   openssl ocsp -issuer csc-prod-chain-ONTAP-Mediator.pem -CAfile csc-
   prod-chain-ONTAP-Mediator.pem -cert csc-prod-ONTAP-Mediator.pem  -url
   ${ocsp_uri} -resp_text -respout resp.der -verify_other csc-prod-
   chain-ONTAP-Mediator.pem
   ```

2. Verify the trust chain of the CSC and expiration dates against the local host:

   ```
   openssl verify
   ```

   The `openssl` version from the PATH must have a valid `cert.pem` (not self-signed).

   ```
   openssl verify -untrusted csc-prod-chain-ONTAP-Mediator.pem -CApath
   ${OPENSSLDIR} csc-prod-ONTAP-Mediator.pem
   openssl verify -untrusted tsa-prod-chain-ONTAP-Mediator.pem -CApath
   ${OPENSSLDIR} tsa-prod-ONTAP-Mediator.pem
   ```

   The command fails if the code-signature check certificate or time-stamp certificate is expired or invalid. Download a newer version of the ONTAP Mediator.

3. Verify the `ontap-mediator-1.13.0.sig.tsr` and `ontap-mediator-1.13.0.tsr` files using the associated certificates:

OpenSSL 3.x

```
openssl ts -verify -data ontap-mediator-1.13.0.sig -in ontap-
mediator-1.13.0.sig.tsr -CAfile tsa-prod-chain-ONTAP-Mediator.pem
-untrusted tsa-prod-ONTAP-Mediator.pem
```

OpenSSL 1.x

```
openssl ts -verify -data ontap-mediator-1.13.0 -in ontap-mediator-
1.13.0.tsr -CAfile tsa-prod-chain-ONTAP-Mediator.pem -partial_chain
```

`.tsr` files contain the time stamp response associated with the installer and the code signature. Processing confirms that the time stamp has a valid signature from TSA and that your input file has not changed.
Your machine performs the verification locally. You do not need to access TSA servers.

4. Verify signatures against the key:

```
openssl -dgst -verify
```

```
openssl dgst -sha256 -verify ONTAP-Mediator-production.pub -signature
ontap-mediator-1.13.0.sig ontap-mediator-1.13.0
```

What's next?

Install the ONTAP Mediator installation package

Install the ONTAP Mediator installation package

To install ONTAP Mediator, you must get the installation package and run the installer on the host.

To upgrade an existing ONTAP Mediator installation, see Upgrade ONTAP Mediator instead.

About this task

- Beginning with ONTAP 9.19.1 and ONTAP Mediator 1.12, you have the option to install ONTAP Mediator with iSCSI and HTTPS support, or support for HTTPS only. If you choose to install ONTAP Mediator with support for HTTPS only, the SCST package is not installed.
- If you upgrade from ONTAP Mediator 1.11 or earlier with SCST installed, and you choose the HTTPS only option during the upgrade, the SCST package is uninstalled and the new ONTAP Mediator version does not contain the SCST package.
- If you plan to use ONTAP Mediator to monitor multiple MetroCluster IP configurations, the support depends on your Mediator type, ONTAP version, and OS. Refer to Use ONTAP Mediator to monitor multiple

MetroCluster IP configurations for more information.

- Installing ONTAP Mediator with support for HTTPS only is supported for the following use cases:
  - You plan to only use ONTAP Mediator to monitor your SnapMirror active sync clusters.
  - You plan to configure ONTAP Mediator with HTTPS to monitor one or more MetroCluster IP configurations.
- ONTAP Mediator must be installed with support for iSCSI and HTTPS in the following scenario:
  - You plan to configure ONTAP Mediator with iSCSI to monitor one or more MetroCluster IP configurations.
- During installation or upgrade to ONTAP Mediator 1.12 or later, the installer prompts you to respond to the following:

```
ONTAP Mediator can be installed with iSCSI and HTTPS support, or HTTPS support
only.
Do you wish to install ONTAP Mediator with iSCSI and HTTPS support?
Y(es)/n(o):
```

  - Respond `Yes` if you plan to use ONTAP Mediator configured with iSCSI to monitor one or more MetroCluster IP configurations.
  - Respond `No` if you plan to only use ONTAP Mediator for SnapMirror active sync, or if you plan to configure ONTAP Mediator with HTTPS to monitor one or more MetroCluster IP configurations.

    Alternatively, you can use the `--https-only` option in the `./ontap-mediator-1.13.0/ontap-mediator-1.13.0` command to specify HTTPS only.
- For automation use cases, you can use the `./ontap-mediator-1.13.0/ontap-mediator-1.13.0 -y` command to skip the installer prompts.

Steps

1. Run the installer and respond to the prompts as required:

```
./ontap-mediator-1.13.0/ontap-mediator-1.13.0
```

```
[root@mediator_host ~]# ./ontap-mediator-1.13.0/ontap-mediator-1.13.0
```

> To skip signature check during installation, use the `./ontap-mediator-1.13.0/ontap-mediator-1.13.0 --skip-code-signature-check` command

The installer creates the required accounts and installs the required packages. If ONTAP Mediator is already installed, the installer prompts you to upgrade instead; for upgrade-specific steps and examples, see Upgrade ONTAP Mediator.

▼ Example ONTAP Mediator installation with HTTPS and iSCSI support (console output)

```
[root@mediator-host ~]# ontap-mediator-1.13.0/ontap-mediator-1.13.0 -y
```

```
ONTAP Mediator: Self Extracting Installer

+ Extracting the ONTAP Mediator installation/upgrade archive
+ Performing the ONTAP Mediator run-time code signature check
   Using openssl from the path: /usr/bin/openssl configured for
CApath:/etc/pki/tls
Error querying OCSP responder
40B70028217F0000:error:1E800080:HTTP
routines:OSSL_HTTP_REQ_CTX_nbio:failed reading
data:crypto/http/http_client.c:596:
40B70028217F0000:error:1E800067:HTTP
routines:OSSL_HTTP_REQ_CTX_exchange:error
receiving:crypto/http/http_client.c:1046:server=http://ocsp.entrust.ne
t:80
   WARNING: The OCSP check failed while attempting to test the Code-
Signature-Check certificate
   Continue without code signature checking (only recommended if
integrity has been established manually)? y(es)/N(o): yes
 SKIPPING: Code signature check, manual override due to lack of OCSP
response
+ Unpacking the ONTAP Mediator installer
+ ONTAP_MEDIATOR_INSTALL_HTTPS_ONLY is set to '0', args are: ''
+ Starting ONTAP Mediator install/update process: previous
installation_mode is ''


ONTAP Mediator can be installed with iSCSI and HTTPS support, or HTTPS
support only.
Do you wish to install ONTAP Mediator with iSCSI and HTTPS support?
Y(es)/n(o): yes
ONTAP Mediator will be installed with mode: ISCSI_HTTPS
+ Selected install package mode: ISCSI_HTTPS


+ Checking that required ports 3260 and 31784 are free
ONTAP Mediator requires two user accounts. One for the service
(netapp), and one for use by ONTAP to the mediator API
(mediatoradmin).
Would you like to use the default account names: netapp +
mediatoradmin? (Y(es)/n(o)): yes




Enter ONTAP Mediator user account (mediatoradmin) password:
```

```
Re-Enter ONTAP Mediator user account (mediatoradmin) password:

+ Checking if SELinux is in enforcing mode
SELinux is set to Enforcing. ONTAP Mediator server requires modifying
the SELinux context of the file
/opt/netapp/lib/ontap_mediator/pyenv/bin/uwsgi from type 'lib_t' to
'bin_t'.
This is neccessary to start the ONTAP Mediator service while SELinux
is set to Enforcing.
Allow SELinux context change?  Y(es)/n(o): yes
The installer will change the SELinux context type of
/opt/netapp/lib/ontap_mediator/pyenv/bin/uwsgi from type 'lib_t' to
'bin_t'.


+ Checking for default Linux firewall
Linux firewall is running. Open ports 31784 and 3260? Y(es)/n(o): yes
success
success
success




+ Installing required packages.


Last metadata expiration check: 0:04:17 ago on Mon 02 Nov 2026
06:27:32 AM EST.
Package openssl-1:3.5.5-2.0.1.el10_2.x86_64 is already installed.
Package which-2.21-44.el10.x86_64 is already installed.
Package libselinux-utils-3.10-1.el10.x86_64 is already installed.
Package perl-Data-Dumper-2.189-512.el10.x86_64 is already installed.
Package bzip2-1.0.8-25.el10.x86_64 is already installed.
Package efibootmgr-18-8.0.1.el10.x86_64 is already installed.
Package mokutil-2:0.7.2-4.el10.x86_64 is already installed.
Package policycoreutils-python-utils-3.10-1.el10.noarch is already
installed.
Package python3-3.12.13-2.0.1.el10_2.x86_64 is already installed.
Dependencies resolved.
================================================================================
================================================================================
=================
 Package                                      Architecture
Version                                     Repository
Size
```

```
================================================================================
================================================================================
=================
Installing:
 elfutils-libelf-devel                          x86_64
0.194-2.el10_2                                 ol10_appstream_netapp
140 k
 gcc                                            x86_64
14.3.1-4.4.0.1.el10                            ol10_appstream_netapp
38 M
 kernel-uek-devel                               x86_64
6.12.0-204.92.4.2.el10uek                      ol10_uek8_netapp
47 M
 make                                           x86_64
1:4.4.1-9.el10                                 ol10_baseos_netapp
620 k
 openssl-devel                                  x86_64
1:3.5.8-2.0.1.el10_2                           ol10_appstream_netapp
3.7 M
 patch                                          x86_64
2.7.6-26.el10                                  ol10_appstream_netapp
133 k
 perl-ExtUtils-MakeMaker                        noarch
2:7.70-513.el10                                ol10_appstream_netapp
344 k
 python3-devel                                  x86_64
3.12.15-1.0.1.el10_2                           ol10_appstream_netapp
521 k
 python3-pip                                    noarch
23.3.2-11.el10_2                               ol10_appstream_netapp
4.4 M
Upgrading:
 openssl                                        x86_64
1:3.5.8-2.0.1.el10_2                           ol10_baseos_netapp
1.4 M
 openssl-libs                                   x86_64
1:3.5.8-2.0.1.el10_2                           ol10_baseos_netapp
2.3 M
 policycoreutils                                x86_64
3.10-2.el10_2                                  ol10_baseos_netapp
568 k
 policycoreutils-python-utils                   noarch
3.10-2.el10_2                                  ol10_appstream_netapp
250 k
 python-unversioned-command                     noarch
3.12.15-1.0.1.el10_2                           ol10_appstream_netapp
```

```
15 k
 python3                                    x86_64
3.12.15-1.0.1.el10_2                       ol10_baseos_netapp
37 k
 python3-libs                               x86_64
3.12.15-1.0.1.el10_2                       ol10_baseos_netapp
11 M
 python3-policycoreutils                    noarch
3.10-2.el10_2                              ol10_appstream_netapp
2.6 M
 which                                      x86_64
2.21-44.el10_0                             ol10_baseos_netapp
62 k
Installing dependencies:
 annobin-docs                               noarch
13.02-2.el10                               ol10_appstream_netapp
147 k
 annobin-plugin-gcc                         x86_64
13.02-2.el10                               ol10_appstream_netapp
1.0 M
 binutils                                   x86_64
2.41-63.0.1.el10                           ol10_baseos_netapp
6.7 M
 binutils-devel                             x86_64
2.41-63.0.1.el10                           ol10_appstream_netapp
8.8 M
 binutils-gold                              x86_64
2.41-63.0.1.el10                           ol10_baseos_netapp
908 k
 cmake-filesystem                           x86_64
3.31.8-1.el10                              ol10_appstream_netapp
57 k
 cpp                                        x86_64
14.3.1-4.4.0.1.el10                        ol10_appstream_netapp
13 M
 dwz                                        x86_64
0.16-1.el10                                ol10_appstream_netapp
150 k
 efi-srpm-macros                            noarch
6-6.0.1.el10                               ol10_appstream_netapp
27 k
 fonts-srpm-macros                          noarch
1:2.0.5-18.el10                            ol10_appstream_netapp
31 k
 forge-srpm-macros                          noarch
0.4.0-6.el10                               ol10_appstream_netapp
```

```
26 k
 gcc-c++                                  x86_64
14.3.1-4.4.0.1.el10                      ol10_appstream_netapp
15 M
 gcc-plugin-annobin                       x86_64
14.3.1-4.4.0.1.el10                      ol10_appstream_netapp
70 k
 glibc-devel                              x86_64
2.39-121.0.1.el10_2                      ol10_appstream_netapp
1.1 M
 go-srpm-macros                           noarch
3.8.0-1.el10                             ol10_appstream_netapp
34 k
 kernel-headers                           x86_64
6.12.0-211.69.1.el10_2                   ol10_appstream_netapp
4.4 M
 kernel-srpm-macros                       noarch
1.0-27.0.1.el10                          ol10_appstream_netapp
12 k
 libstdc++-devel                          x86_64
14.3.1-4.4.0.1.el10                      ol10_appstream_netapp
3.5 M
 libxcrypt-devel                          x86_64
4.4.36-10.el10                           ol10_appstream_netapp
42 k
 libzstd-devel                            x86_64
1.5.5-9.el10                             ol10_appstream_netapp
55 k
 lua-srpm-macros                          noarch
1-15.el10                                ol10_appstream_netapp
8.7 k
 ocaml-srpm-macros                        noarch
10-4.el10                                ol10_appstream_netapp
9.0 k
 openblas-srpm-macros                     noarch
2-19.el10                                ol10_appstream_netapp
7.6 k
 package-notes-srpm-macros                noarch
0.5-13.el10                              ol10_appstream_netapp
9.1 k
 perl-AutoSplit                           noarch
5.74-515.el10_2                          ol10_appstream_netapp
103 k
 perl-Benchmark                           noarch
1.25-515.el10_2                          ol10_appstream_netapp
108 k
```

```
 perl-CPAN-Meta-Requirements                  noarch
2.143-11.el10                               ol10_appstream_netapp
43 k
 perl-CPAN-Meta-YAML                          noarch
0.018-512.el10                              ol10_appstream_netapp
31 k
 perl-Devel-PPPort                            x86_64
3.72-512.el10                               ol10_appstream_netapp
227 k
 perl-ExtUtils-Command                        noarch
2:7.70-513.el10                             ol10_appstream_netapp
14 k
 perl-ExtUtils-Constant                       noarch
0.25-515.el10_2                             ol10_appstream_netapp
134 k
 perl-ExtUtils-Install                        noarch
2.22-511.el10                               ol10_appstream_netapp
51 k
 perl-ExtUtils-Manifest                       noarch
1:1.75-511.el10                             ol10_appstream_netapp
40 k
 perl-ExtUtils-ParseXS                        noarch
1:3.51-512.el10                             ol10_appstream_netapp
211 k
 perl-File-Compare                            noarch
1.100.800-515.el10_2                        ol10_appstream_netapp
94 k
 perl-File-Copy                               noarch
2.41-515.el10_2                             ol10_appstream_netapp
101 k
 perl-I18N-Langinfo                           x86_64
0.24-515.el10_2                             ol10_appstream_netapp
107 k
 perl-JSON-PP                                 noarch
1:4.16-513.el10                             ol10_appstream_netapp
86 k
 perl-Test-Harness                            noarch
1:3.48-512.el10                             ol10_appstream_netapp
375 k
 perl-lib                                     x86_64
0.65-515.el10_2                             ol10_appstream_netapp
96 k
 perl-srpm-macros                             noarch
1-57.el10                                   ol10_appstream_netapp
8.4 k
 perl-version                                 x86_64
```

```
8:0.99.32-4.el10                                ol10_appstream_netapp
74 k
 pyproject-srpm-macros                            noarch
1.18.5-1.el10                                   ol10_appstream_netapp
26 k
 python-srpm-macros                               noarch
3.12-11.el10                                    ol10_appstream_netapp
47 k
 qt6-srpm-macros                                  noarch
6.10.1-1.el10                                   ol10_appstream_netapp
12 k
 redhat-rpm-config                                noarch
295-1.0.1.el10                                  ol10_appstream_netapp
155 k
 rust-toolset-srpm-macros                         noarch
1.92.0-2.el10_2                                 ol10_appstream_netapp
27 k
 systemtap-sdt-devel                              x86_64
5.4-3.0.1.el10                                  ol10_appstream_netapp
85 k
 systemtap-sdt-dtrace                             x86_64
5.4-3.0.1.el10                                  ol10_appstream_netapp
84 k
 zlib-ng-compat-devel                             x86_64
2.2.3-3.el10_1                                  ol10_appstream_netapp
45 k
Installing weak dependencies:
 perl-CPAN-Meta                                   noarch
2.150010-511.el10                               ol10_appstream_netapp
288 k
 perl-Encode-Locale                               noarch
1.05-31.el10                                    ol10_appstream_netapp
22 k
 perl-Time-HiRes                                  x86_64
4:1.9777-511.el10                               ol10_appstream_netapp
64 k
 perl-devel                                       x86_64
4:5.40.2-512.2.el10_0                           ol10_appstream_netapp
925 k
 perl-doc                                         noarch
5.40.2-512.2.el10_0                             ol10_appstream_netapp
5.3 M

Transaction Summary
================================================================================
================================================================================
```

```
==================
Install  64 Packages
Upgrade   9 Packages

Total download size: 177 M
Is this ok [y/N]: y
Downloading Packages:
(1/73): make-4.4.1-9.el10.x86_64.rpm
3.4 MB/s | 620 kB     00:00
(2/73): binutils-gold-2.41-63.0.1.el10.x86_64.rpm
4.3 MB/s | 908 kB     00:00
(3/73): annobin-docs-13.02-2.el10.noarch.rpm
2.2 MB/s | 147 kB     00:00
(4/73): annobin-plugin-gcc-13.02-2.el10.x86_64.rpm
5.8 MB/s | 1.0 MB     00:00
(5/73): cmake-filesystem-3.31.8-1.el10.x86_64.rpm
862 kB/s |  57 kB     00:00
(6/73): binutils-2.41-63.0.1.el10.x86_64.rpm
11 MB/s | 6.7 MB     00:00
(7/73): dwz-0.16-1.el10.x86_64.rpm
937 kB/s | 150 kB     00:00
(8/73): binutils-devel-2.41-63.0.1.el10.x86_64.rpm
14 MB/s | 8.8 MB     00:00
(9/73): efi-srpm-macros-6-6.0.1.el10.noarch.rpm
221 kB/s |  27 kB     00:00
(10/73): elfutils-libelf-devel-0.194-2.el10_2.x86_64.rpm
1.9 MB/s | 140 kB     00:00
(11/73): fonts-srpm-macros-2.0.5-18.el10.noarch.rpm
402 kB/s |  31 kB     00:00
(12/73): cpp-14.3.1-4.4.0.1.el10.x86_64.rpm
21 MB/s |  13 MB     00:00
(13/73): forge-srpm-macros-0.4.0-6.el10.noarch.rpm
234 kB/s |  26 kB     00:00
(14/73): gcc-plugin-annobin-14.3.1-4.4.0.1.el10.x86_64.rpm
1.2 MB/s |  70 kB     00:00
(15/73): glibc-devel-2.39-121.0.1.el10_2.x86_64.rpm
6.0 MB/s | 1.1 MB     00:00
(16/73): go-srpm-macros-3.8.0-1.el10.noarch.rpm
701 kB/s |  34 kB     00:00
(17/73): gcc-c++-14.3.1-4.4.0.1.el10.x86_64.rpm
21 MB/s |  15 MB     00:00
(18/73): kernel-srpm-macros-1.0-27.0.1.el10.noarch.rpm
419 kB/s |  12 kB     00:00
(19/73): libstdc++-devel-14.3.1-4.4.0.1.el10.x86_64.rpm
19 MB/s | 3.5 MB     00:00
(20/73): gcc-14.3.1-4.4.0.1.el10.x86_64.rpm
```

```
29 MB/s |  38 MB     00:01
(21/73): libxcrypt-devel-4.4.36-10.el10.x86_64.rpm
145 kB/s |  42 kB     00:00
(22/73): lua-srpm-macros-1-15.el10.noarch.rpm
277 kB/s | 8.7 kB     00:00
(23/73): libzstd-devel-1.5.5-9.el10.x86_64.rpm
824 kB/s |  55 kB     00:00
(24/73): ocaml-srpm-macros-10-4.el10.noarch.rpm
194 kB/s | 9.0 kB     00:00
(25/73): openblas-srpm-macros-2-19.el10.noarch.rpm
68 kB/s | 7.6 kB     00:00
(26/73): package-notes-srpm-macros-0.5-13.el10.noarch.rpm
102 kB/s | 9.1 kB     00:00
(27/73): openssl-devel-3.5.8-2.0.1.el10_2.x86_64.rpm
14 MB/s | 3.7 MB     00:00
(28/73): patch-2.7.6-26.el10.x86_64.rpm
1.3 MB/s | 133 kB     00:00
(29/73): perl-AutoSplit-5.74-515.el10_2.noarch.rpm
1.6 MB/s | 103 kB     00:00
(30/73): perl-Benchmark-1.25-515.el10_2.noarch.rpm
1.5 MB/s | 108 kB     00:00
(31/73): perl-CPAN-Meta-2.150010-511.el10.noarch.rpm
4.0 MB/s | 288 kB     00:00
(32/73): perl-CPAN-Meta-YAML-0.018-512.el10.noarch.rpm
759 kB/s |  31 kB     00:00
(33/73): perl-CPAN-Meta-Requirements-2.143-11.el10.noarch.rpm
471 kB/s |  43 kB     00:00
(34/73): perl-Devel-PPPort-3.72-512.el10.x86_64.rpm
3.0 MB/s | 227 kB     00:00
(35/73): perl-Encode-Locale-1.05-31.el10.noarch.rpm
195 kB/s |  22 kB     00:00
(36/73): perl-ExtUtils-Command-7.70-513.el10.noarch.rpm
201 kB/s |  14 kB     00:00
(37/73): perl-ExtUtils-Constant-0.25-515.el10_2.noarch.rpm
1.8 MB/s | 134 kB     00:00
(38/73): perl-ExtUtils-Install-2.22-511.el10.noarch.rpm
971 kB/s |  51 kB     00:00
(39/73): perl-ExtUtils-Manifest-1.75-511.el10.noarch.rpm
893 kB/s |  40 kB     00:00
(40/73): perl-ExtUtils-MakeMaker-7.70-513.el10.noarch.rpm
3.9 MB/s | 344 kB     00:00
(41/73): perl-ExtUtils-ParseXS-3.51-512.el10.noarch.rpm
4.0 MB/s | 211 kB     00:00
(42/73): perl-File-Compare-1.100.800-515.el10_2.noarch.rpm
1.7 MB/s |  94 kB     00:00
(43/73): perl-File-Copy-2.41-515.el10_2.noarch.rpm
```

```
1.0 MB/s | 101 kB     00:00
(44/73): perl-I18N-Langinfo-0.24-515.el10_2.x86_64.rpm
1.3 MB/s | 107 kB     00:00
(45/73): perl-JSON-PP-4.16-513.el10.noarch.rpm
1.1 MB/s |  86 kB     00:00
(46/73): perl-Test-Harness-3.48-512.el10.noarch.rpm
4.6 MB/s | 375 kB     00:00
(47/73): perl-Time-HiRes-1.9777-511.el10.x86_64.rpm
758 kB/s |  64 kB     00:00
(48/73): perl-devel-5.40.2-512.2.el10_0.x86_64.rpm
6.0 MB/s | 925 kB     00:00
(49/73): perl-lib-0.65-515.el10_2.x86_64.rpm
1.1 MB/s |  96 kB     00:00
(50/73): perl-srpm-macros-1-57.el10.noarch.rpm
178 kB/s | 8.4 kB     00:00
(51/73): perl-doc-5.40.2-512.2.el10_0.noarch.rpm
14 MB/s | 5.3 MB     00:00
(52/73): perl-version-0.99.32-4.el10.x86_64.rpm
478 kB/s |  74 kB     00:00
(53/73): pyproject-srpm-macros-1.18.5-1.el10.noarch.rpm
446 kB/s |  26 kB     00:00
(54/73): python-srpm-macros-3.12-11.el10.noarch.rpm
616 kB/s |  47 kB     00:00
(55/73): python3-devel-3.12.15-1.0.1.el10_2.x86_64.rpm
4.3 MB/s | 521 kB     00:00
(56/73): qt6-srpm-macros-6.10.1-1.el10.noarch.rpm
256 kB/s |  12 kB     00:00
(57/73): redhat-rpm-config-295-1.0.1.el10.noarch.rpm
1.9 MB/s | 155 kB     00:00
(58/73): python3-pip-23.3.2-11.el10_2.noarch.rpm
13 MB/s | 4.4 MB     00:00
(59/73): rust-toolset-srpm-macros-1.92.0-2.el10_2.noarch.rpm
247 kB/s |  27 kB     00:00
(60/73): systemtap-sdt-dtrace-5.4-3.0.1.el10.x86_64.rpm
1.0 MB/s |  84 kB     00:00
(61/73): systemtap-sdt-devel-5.4-3.0.1.el10.x86_64.rpm
704 kB/s |  85 kB     00:00
(62/73): zlib-ng-compat-devel-2.2.3-3.el10_1.x86_64.rpm
448 kB/s |  45 kB     00:00
(63/73): openssl-3.5.8-2.0.1.el10_2.x86_64.rpm
9.0 MB/s | 1.4 MB     00:00
(64/73): openssl-libs-3.5.8-2.0.1.el10_2.x86_64.rpm
7.9 MB/s | 2.3 MB     00:00
(65/73): policycoreutils-3.10-2.el10_2.x86_64.rpm
3.8 MB/s | 568 kB     00:00
(66/73): python3-3.12.15-1.0.1.el10_2.x86_64.rpm
```

```
352 kB/s |  37 kB     00:00
(67/73): kernel-uek-devel-6.12.0-204.92.4.2.el10uek.x86_64.rpm
33 MB/s |  47 MB     00:01
(68/73): kernel-headers-6.12.0-211.69.1.el10_2.x86_64.rpm
1.0 MB/s | 4.4 MB     00:04
(69/73): which-2.21-44.el10_0.x86_64.rpm
814 kB/s |  62 kB     00:00
(70/73): python3-libs-3.12.15-1.0.1.el10_2.x86_64.rpm
13 MB/s |  11 MB     00:00
(71/73): python-unversioned-command-3.12.15-1.0.1.el10_2.noarch.rpm
113 kB/s |  15 kB     00:00
(72/73): policycoreutils-python-utils-3.10-2.el10_2.noarch.rpm
1.7 MB/s | 250 kB     00:00
(73/73): python3-policycoreutils-3.10-2.el10_2.noarch.rpm
11 MB/s | 2.6 MB     00:00
--------------------------------------------------------------------------
--------------------------------------------------------------------------
-----------------
Total
28 MB/s | 177 MB     00:06
Running transaction check
Transaction check succeeded.
Running transaction test
Transaction test succeeded.
Running transaction
  Preparing        :
1/1
  Installing       : binutils-gold-2.41-63.0.1.el10.x86_64
1/82
  Running scriptlet: binutils-gold-2.41-63.0.1.el10.x86_64
1/82
  Installing       : binutils-2.41-63.0.1.el10.x86_64
2/82
  Running scriptlet: binutils-2.41-63.0.1.el10.x86_64
2/82
  Upgrading        : openssl-libs-1:3.5.8-2.0.1.el10_2.x86_64
3/82
  Installing       : perl-version-8:0.99.32-4.el10.x86_64
4/82
  Installing       : perl-File-Copy-2.41-515.el10_2.noarch
5/82
  Installing       : perl-CPAN-Meta-Requirements-2.143-11.el10.noarch
6/82
  Upgrading        : python-unversioned-command-3.12.15-
1.0.1.el10_2.noarch
7/82
```

```
  Upgrading        : python3-3.12.15-1.0.1.el10_2.x86_64
8/82
  Upgrading        : python3-libs-3.12.15-1.0.1.el10_2.x86_64
9/82
  Installing       : systemtap-sdt-dtrace-5.4-3.0.1.el10.x86_64
10/82
  Installing       : perl-Time-HiRes-4:1.9777-511.el10.x86_64
11/82
  Installing       : perl-JSON-PP-1:4.16-513.el10.noarch
12/82
  Installing       : perl-File-Compare-1.100.800-515.el10_2.noarch
13/82
  Installing       : perl-ExtUtils-ParseXS-1:3.51-512.el10.noarch
14/82
  Installing       : systemtap-sdt-devel-5.4-3.0.1.el10.x86_64
15/82
  Installing       : python3-pip-23.3.2-11.el10_2.noarch
16/82
  Installing       : perl-ExtUtils-Command-2:7.70-513.el10.noarch
17/82
  Installing       : perl-ExtUtils-Manifest-1:1.75-511.el10.noarch
18/82
  Upgrading        : policycoreutils-3.10-2.el10_2.x86_64
19/82
  Running scriptlet: policycoreutils-3.10-2.el10_2.x86_64
19/82
  Upgrading        : python3-policycoreutils-3.10-2.el10_2.noarch
20/82
  Installing       : rust-toolset-srpm-macros-1.92.0-2.el10_2.noarch
21/82
  Installing       : qt6-srpm-macros-6.10.1-1.el10.noarch
22/82
  Installing       : pyproject-srpm-macros-1.18.5-1.el10.noarch
23/82
  Installing       : perl-srpm-macros-1-57.el10.noarch
24/82
  Installing       : perl-lib-0.65-515.el10_2.x86_64
25/82
  Installing       : perl-doc-5.40.2-512.2.el10_0.noarch
26/82
  Installing       : perl-I18N-Langinfo-0.24-515.el10_2.x86_64
27/82
  Installing       : perl-Encode-Locale-1.05-31.el10.noarch
28/82
  Installing       : perl-ExtUtils-Constant-0.25-515.el10_2.noarch
29/82
```

```
  Installing       : perl-Devel-PPPort-3.72-512.el10.x86_64
30/82
  Installing       : perl-CPAN-Meta-YAML-0.018-512.el10.noarch
31/82
  Installing       : perl-CPAN-Meta-2.150010-511.el10.noarch
32/82
  Installing       : perl-Benchmark-1.25-515.el10_2.noarch
33/82
  Installing       : perl-Test-Harness-1:3.48-512.el10.noarch
34/82
  Installing       : perl-AutoSplit-5.74-515.el10_2.noarch
35/82
  Installing       : package-notes-srpm-macros-0.5-13.el10.noarch
36/82
  Installing       : openblas-srpm-macros-2-19.el10.noarch
37/82
  Installing       : ocaml-srpm-macros-10-4.el10.noarch
38/82
  Installing       : lua-srpm-macros-1-15.el10.noarch
39/82
  Installing       : libzstd-devel-1.5.5-9.el10.x86_64
40/82
  Installing       : libstdc++-devel-14.3.1-4.4.0.1.el10.x86_64
41/82
  Installing       : kernel-srpm-macros-1.0-27.0.1.el10.noarch
42/82
  Installing       : kernel-headers-6.12.0-211.69.1.el10_2.x86_64
43/82
  Installing       : libxcrypt-devel-4.4.36-10.el10.x86_64
44/82
  Installing       : glibc-devel-2.39-121.0.1.el10_2.x86_64
45/82
  Installing       : efi-srpm-macros-6-6.0.1.el10.noarch
46/82
  Installing       : dwz-0.16-1.el10.x86_64
47/82
  Installing       : cpp-14.3.1-4.4.0.1.el10.x86_64
48/82
  Installing       : cmake-filesystem-3.31.8-1.el10.x86_64
49/82
  Installing       : zlib-ng-compat-devel-2.2.3-3.el10_1.x86_64
50/82
  Installing       : binutils-devel-2.41-63.0.1.el10.x86_64
51/82
  Installing       : annobin-docs-13.02-2.el10.noarch
52/82
```

```
  Installing       : make-1:4.4.1-9.el10.x86_64
53/82
  Installing       : gcc-14.3.1-4.4.0.1.el10.x86_64
54/82
  Installing       : annobin-plugin-gcc-13.02-2.el10.x86_64
55/82
  Installing       : gcc-c++-14.3.1-4.4.0.1.el10.x86_64
56/82
  Installing       : gcc-plugin-annobin-14.3.1-4.4.0.1.el10.x86_64
57/82
  Installing       : fonts-srpm-macros-1:2.0.5-18.el10.noarch
58/82
  Installing       : forge-srpm-macros-0.4.0-6.el10.noarch
59/82
  Installing       : go-srpm-macros-3.8.0-1.el10.noarch
60/82
  Installing       : python-srpm-macros-3.12-11.el10.noarch
61/82
  Installing       : redhat-rpm-config-295-1.0.1.el10.noarch
62/82
  Running scriptlet: redhat-rpm-config-295-1.0.1.el10.noarch
62/82
  Installing       : perl-devel-4:5.40.2-512.2.el10_0.x86_64
63/82
  Installing       : perl-ExtUtils-Install-2.22-511.el10.noarch
64/82
  Installing       : perl-ExtUtils-MakeMaker-2:7.70-513.el10.noarch
65/82
  Installing       : kernel-uek-devel-6.12.0-204.92.4.2.el10uek.x86_64
66/82
  Running scriptlet: kernel-uek-devel-6.12.0-204.92.4.2.el10uek.x86_64
66/82
  Installing       : elfutils-libelf-devel-0.194-2.el10_2.x86_64
67/82
  Upgrading        : policycoreutils-python-utils-3.10-2.el10_2.noarch
68/82
  Installing       : python3-devel-3.12.15-1.0.1.el10_2.x86_64
69/82
  Installing       : openssl-devel-1:3.5.8-2.0.1.el10_2.x86_64
70/82
  Upgrading        : openssl-1:3.5.8-2.0.1.el10_2.x86_64
71/82
  Upgrading        : which-2.21-44.el10_0.x86_64
72/82
  Installing       : patch-2.7.6-26.el10.x86_64
73/82
```

```
  Cleanup          : openssl-1:3.5.5-2.0.1.el10_2.x86_64
74/82
  Cleanup          : policycoreutils-python-utils-3.10-1.el10.noarch
75/82
  Cleanup          : python3-policycoreutils-3.10-1.el10.noarch
76/82
  Cleanup          : python3-3.12.13-2.0.1.el10_2.x86_64
77/82
  Cleanup          : python-unversioned-command-3.12.13-
2.0.1.el10_2.noarch
78/82
  Cleanup          : python3-libs-3.12.13-2.0.1.el10_2.x86_64
79/82
  Cleanup          : openssl-libs-1:3.5.5-2.0.1.el10_2.x86_64
80/82
  Running scriptlet: policycoreutils-3.10-1.el10.x86_64
81/82
  Cleanup          : policycoreutils-3.10-1.el10.x86_64
81/82
  Cleanup          : which-2.21-44.el10.x86_64
82/82
  Running scriptlet: which-2.21-44.el10.x86_64
82/82

Upgraded:
  openssl-1:3.5.8-2.0.1.el10_2.x86_64                  openssl-libs-
1:3.5.8-2.0.1.el10_2.x86_64                  policycoreutils-3.10-
2.el10_2.x86_64
  policycoreutils-python-utils-3.10-2.el10_2.noarch     python-
unversioned-command-3.12.15-1.0.1.el10_2.noarch     python3-3.12.15-
1.0.1.el10_2.x86_64
  python3-libs-3.12.15-1.0.1.el10_2.x86_64             python3-
policycoreutils-3.10-2.el10_2.noarch               which-2.21-
44.el10_0.x86_64
Installed:
  annobin-docs-13.02-2.el10.noarch                    annobin-plugin-
gcc-13.02-2.el10.x86_64               binutils-2.41-63.0.1.el10.x86_64
  binutils-devel-2.41-63.0.1.el10.x86_64              binutils-gold-
2.41-63.0.1.el10.x86_64               cmake-filesystem-3.31.8-
1.el10.x86_64
  cpp-14.3.1-4.4.0.1.el10.x86_64                      dwz-0.16-
1.el10.x86_64                                efi-srpm-macros-6-
6.0.1.el10.noarch
  elfutils-libelf-devel-0.194-2.el10_2.x86_64         fonts-srpm-
macros-1:2.0.5-18.el10.noarch             forge-srpm-macros-0.4.0-
6.el10.noarch
```

```
  gcc-14.3.1-4.4.0.1.el10.x86_64                    gcc-c++-14.3.1-
4.4.0.1.el10.x86_64                  gcc-plugin-annobin-14.3.1-
4.4.0.1.el10.x86_64
  glibc-devel-2.39-121.0.1.el10_2.x86_64            go-srpm-macros-
3.8.0-1.el10.noarch                  kernel-headers-6.12.0-
211.69.1.el10_2.x86_64
  kernel-srpm-macros-1.0-27.0.1.el10.noarch         kernel-uek-
devel-6.12.0-204.92.4.2.el10uek.x86_64    libstdc++-devel-14.3.1-
4.4.0.1.el10.x86_64
  libxcrypt-devel-4.4.36-10.el10.x86_64             libzstd-devel-
1.5.5-9.el10.x86_64                   lua-srpm-macros-1-
15.el10.noarch
  make-1:4.4.1-9.el10.x86_64                        ocaml-srpm-
macros-10-4.el10.noarch                  openblas-srpm-macros-2-
19.el10.noarch
  openssl-devel-1:3.5.8-2.0.1.el10_2.x86_64         package-notes-
srpm-macros-0.5-13.el10.noarch         patch-2.7.6-26.el10.x86_64
  perl-AutoSplit-5.74-515.el10_2.noarch             perl-Benchmark-
1.25-515.el10_2.noarch               perl-CPAN-Meta-2.150010-
511.el10.noarch
  perl-CPAN-Meta-Requirements-2.143-11.el10.noarch    perl-CPAN-Meta-
YAML-0.018-512.el10.noarch           perl-Devel-PPPort-3.72-
512.el10.x86_64
  perl-Encode-Locale-1.05-31.el10.noarch            perl-ExtUtils-
Command-2:7.70-513.el10.noarch         perl-ExtUtils-Constant-0.25-
515.el10_2.noarch
  perl-ExtUtils-Install-2.22-511.el10.noarch        perl-ExtUtils-
MakeMaker-2:7.70-513.el10.noarch       perl-ExtUtils-Manifest-1:1.75-
511.el10.noarch
  perl-ExtUtils-ParseXS-1:3.51-512.el10.noarch      perl-File-
Compare-1.100.800-515.el10_2.noarch        perl-File-Copy-2.41-
515.el10_2.noarch
  perl-I18N-Langinfo-0.24-515.el10_2.x86_64         perl-JSON-PP-
1:4.16-513.el10.noarch                 perl-Test-Harness-1:3.48-
512.el10.noarch
  perl-Time-HiRes-4:1.9777-511.el10.x86_64          perl-devel-
4:5.40.2-512.2.el10_0.x86_64             perl-doc-5.40.2-
512.2.el10_0.noarch
  perl-lib-0.65-515.el10_2.x86_64                   perl-srpm-
macros-1-57.el10.noarch                  perl-version-8:0.99.32-
4.el10.x86_64
  pyproject-srpm-macros-1.18.5-1.el10.noarch        python-srpm-
macros-3.12-11.el10.noarch             python3-devel-3.12.15-
1.0.1.el10_2.x86_64
  python3-pip-23.3.2-11.el10_2.noarch               qt6-srpm-macros-
6.10.1-1.el10.noarch               redhat-rpm-config-295-
```

```
1.0.1.el10.noarch
  rust-toolset-srpm-macros-1.92.0-2.el10_2.noarch     systemtap-sdt-
devel-5.4-3.0.1.el10.x86_64            systemtap-sdt-dtrace-5.4-
3.0.1.el10.x86_64
  zlib-ng-compat-devel-2.2.3-3.el10_1.x86_64

Complete!
OS package installations finished
+ Installing ONTAP Mediator. (Log: /root/ontap_mediator.26mL7P/ontap-
mediator-1.13.0/ontap-mediator-1.13.0/install_20261102063122.log)
    This step will take several minutes. Use the log file to view
progress.



    Sudoer config verified

    ONTAP Mediator rsyslog and logging rotation enabled
+ Install successful. (Moving log to
/opt/netapp/lib/ontap_mediator/log/install_20261102063122.log)
+ WARNING: This system supports UEFI
           Secure Boot (SB) is currently enabled on this system.
           The following action need be taken:
           Using the keys in
/opt/netapp/lib/ontap_mediator/ontap_mediator/SCST_mod_keys follow
           instructions in
/opt/netapp/lib/ontap_mediator/ontap_mediator/SCST_mod_keys/README.mod
ule-signing
           to sign the SCST kernel module. Note that reboot will be
needed.
     SCST will not start automatically when Secure Boot is enabled and
not configured properly.

+ Note: ONTAP Mediator generated a self-signed server certificate for
temporary use on
    this host. If the DNS name or IP address for the host is changed,
the certificate
    will no longer be valid. The default certificates should be
replaced with secure
    trusted certificates signed by a known certificate authority prior
to use for production.
    For more information, see /opt/netapp/lib/ontap_mediator/README

+ Note: ONTAP Mediator uses a kernel module compiled specifically for
the current
        OS. Using 'yum update' to upgrade the kernel might cause
```

```
service interruption.
    For more information, see /opt/netapp/lib/ontap_mediator/README
[root@mediator-host ~]#
```

▼ **Example ONTAP Mediator installation with support for HTTPS only (console output)**

```
[root@localhost ~]# ontap-mediator-1.13.0/ontap-mediator-1.13.0
--https-only

ONTAP Mediator: Self Extracting Installer

+ Extracting the ONTAP Mediator installation/upgrade archive
+ Performing the ONTAP Mediator run-time code signature check
   Using openssl from the path: /usr/bin/openssl configured for
CApath:/etc/pki/tls
Error querying OCSP responder
4097E2673F7F0000:error:10080002:BIO routines:BIO_lookup_ex:system
lib:crypto/bio/bio_addr.c:767:Name or service not known
   WARNING: The OCSP check failed while attempting to test the Code-
Signature-Check certificate
   Continue without code signature checking (only recommended if
integrity has been established manually)? y(es)/N(o): yes
 SKIPPING: Code signature check, manual override due to lack of OCSP
response
+ Unpacking the ONTAP Mediator installer
+ ONTAP_MEDIATOR_INSTALL_HTTPS_ONLY is set to '1', args are: ''
+ Starting ONTAP Mediator install/update process: previous
installation_mode is ''
+ Selected install package mode: HTTPS_ONLY


+ Checking that required port 31784 is free
ONTAP Mediator requires two user accounts. One for the service
(netapp), and one for use by ONTAP to the mediator API
(mediatoradmin).
Would you like to use the default account names: netapp +
mediatoradmin? (Y(es)/n(o)): yes




Enter ONTAP Mediator user account (mediatoradmin) password:

Re-Enter ONTAP Mediator user account (mediatoradmin) password:

+ Checking if SELinux is in enforcing mode
```

```
SELinux is set to Enforcing. ONTAP Mediator server requires modifying
the SELinux context of the file
/opt/netapp/lib/ontap_mediator/pyenv/bin/uwsgi from type 'lib_t' to
'bin_t'.
This is neccessary to start the ONTAP Mediator service while SELinux
is set to Enforcing.
Allow SELinux context change?  Y(es)/n(o): yes
The installer will change the SELinux context type of
/opt/netapp/lib/ontap_mediator/pyenv/bin/uwsgi from type 'lib_t' to
'bin_t'.


+ Checking for default Linux firewall




+ Installing required packages.


Updating Subscription Management repositories.
Unable to read consumer identity

This system is not registered with an entitlement server. You can use
"rhc" or "subscription-manager" to register.

Last metadata expiration check: 1:00:07 ago on Fri 16 Oct 2026
01:18:47 AM EDT.
Package openssl-1:3.5.8-2.el9_8.x86_64 is already installed.
Package openssl-devel-1:3.5.8-2.el9_8.x86_64 is already installed.
Package gcc-11.5.0-15.el9.x86_64 is already installed.
Package make-1:4.3-8.el9.x86_64 is already installed.
Package which-2.21-30.el9.x86_64 is already installed.
Package libselinux-utils-3.6-4.el9.x86_64 is already installed.
Package patch-2.7.6-16.el9.x86_64 is already installed.
Package perl-Data-Dumper-2.174-462.el9.x86_64 is already installed.
Package perl-ExtUtils-MakeMaker-2:7.60-3.el9.noarch is already
installed.
Package bzip2-1.0.8-11.el9.x86_64 is already installed.
Package efibootmgr-16-12.el9.x86_64 is already installed.
Package mokutil-2:0.7.2-6.el9_9.x86_64 is already installed.
Package python3-pip-21.3.1-2.el9_8.noarch is already installed.
Package elfutils-libelf-devel-0.195-1.el9.x86_64 is already installed.
Package policycoreutils-python-utils-3.6-9.el9.noarch is already
installed.
Package python3.12-3.12.15-1.el9_8.x86_64 is already installed.
```

```
Package python3.12-devel-3.12.15-1.el9_8.x86_64 is already installed.
Dependencies resolved.
Nothing to do.
Complete!
OS package installations finished
+ Installing ONTAP Mediator. (Log: /root/ontap_mediator.aq18RI/ontap-
mediator-1.13.0/ontap-mediator-1.13.0/install_20261016021831.log)
    This step will take several minutes. Use the log file to view
progress.
    Sudoer config verified
    ONTAP Mediator rsyslog and logging rotation enabled
+ Install successful. (Moving log to
/opt/netapp/lib/ontap_mediator/log/install_20261016021831.log)
+ WARNING: This system supports UEFI
+ Note: Not signing the SCST modules since they were not installed by
this installation.
+ Note: SCST will not start automatically if Secure Boot is enabled
and not configured properly.

+ Note: ONTAP Mediator generated a self-signed server certificate for
temporary use on
    this host. If the DNS name or IP address for the host is changed,
the certificate
    will no longer be valid. The default certificates should be
replaced with secure
    trusted certificates signed by a known certificate authority prior
to use for production.
    For more information, see /opt/netapp/lib/ontap_mediator/README
[root@localhost ~]#
[root@localhost ~]#
```

Register the security key for UEFI Secure Boot

Beginning with ONTAP Mediator 1.4, the Secure Boot mechanism is enabled on UEFI systems. When Secure Boot is enabled, you must take additional steps to register the security key after installation.

> If you installed ONTAP Mediator 1.12 or later with support for HTTPS only, the SCST package is not installed and you can skip this task.

Steps

1. Follow the instructions in the README file to sign the SCST kernel module:

```
/opt/netapp/lib/ontap_mediator/ontap_mediator/SCST_mod_keys/README.modul
e-signing
```

2. Locate the required keys:

```
/opt/netapp/lib/ontap_mediator/ontap_mediator/SCST_mod_keys
```

After installation, the system output provides the README files and key location.

3. Add the public key to the MOK list:

```
mokutil --import
/opt/netapp/lib/ontap_mediator/ontap_mediator/SCST_mod_keys/scst_module_
key.der
```

You can leave the private key in its default location or move it to a secure location. You must keep the public key in its existing location so the Boot Manager can use it. For more information, see the README.module-signing file:

```
[root@hostname ~]# ls
/opt/netapp/lib/ontap_mediator/ontap_mediator/SCST_mod_keys/
README.module-signing scst_module_key.der scst_module_key.priv
```

4. Reboot the host and use your device's UEFI Boot Manager to approve the new MOK. You'll need the passphrase provided for the `mokutil` utility in Install ONTAP Mediator when UEFI Secure Boot is enabled.

Sign SCST kernel modules

After ONTAP Mediator is installed, if the systemctl status `mediator-scst` shows as failed (inactive), follow these steps to sign the SCST kernel module.

If you installed ONTAP Mediator 1.12 or later with support for HTTPS only, the SCST package is not installed and you can skip this task.

Steps

1. During the build process a public/private key pair is generated in the `/opt/netapp/lib/ontap_mediator/ontap_mediator/SCST_mod_keys/` directory. Run the following command to view the generated keys:

```
ls /opt/netapp/lib/ontap_mediator/ontap_mediator/SCST_mod_keys/
```

```
README.module-signing  scst_module_key.der  scst_module_key.priv
```

2. Start the process of importing the public key into the UEFI key repository by running the following command:

```
mokutil --import
```

```
/opt/netapp/lib/ontap_mediator/ontap_mediator/SCST_mod_keys/scst_module_
key.der
```

The `mokutil` utility prompts you to enter a password twice.

3. The mokutil software requests a temporary password to be used for this key during the import process.
4. Verify whether the import process has started with the following command, and then reboot the system. The bootloader starts the EFI MOK manager.

```
mokutil --list-new
```

5. Use the menus on the screen to turn on the SCST kernel module key. After booting, run the following command. Once the service starts, the SCST kernel modules are signed.

```
systemctl status mediator-scst
```

What's next?

Verify the ONTAP Mediator installation status

Verify the ONTAP Mediator installation status

After you install ONTAP Mediator, verify that it is running successfully.

Steps

1. View the status of ONTAP Mediator:

```
systemctl status ontap_mediator
```

```
 ontap_mediator.service - ONTAP Mediator
Loaded: loaded (/etc/systemd/system/ontap_mediator.service; enabled;
vendor preset: disabled)
Active: active (running) since Mon 2022-04-18 10:41:49 EDT; 1 weeks 0
days ago
Process: 286710 ExecStop=/bin/kill -s INT $MAINPID (code=exited,
status=0/SUCCESS)
Main PID: 286712 (uwsgi)
Status: "uWSGI is ready"
Tasks: 3 (limit: 49473)
Memory: 139.2M
CGroup: /system.slice/ontap_mediator.service
      ├─286712 /opt/netapp/lib/ontap_mediator/pyenv/bin/uwsgi --ini
/opt/netapp/lib/ontap_mediator/uwsgi/ontap_mediator.ini
      ├─286716 /opt/netapp/lib/ontap_mediator/pyenv/bin/uwsgi --ini
```

```
/opt/netapp/lib/ontap_mediator/uwsgi/ontap_mediator.ini
      └─286717 /opt/netapp/lib/ontap_mediator/pyenv/bin/uwsgi --ini
/opt/netapp/lib/ontap_mediator/uwsgi/ontap_mediator.ini
```

2. View the SCST status:

```
systemctl status mediator-scst
```

> If you installed ONTAP Mediator 1.12 or later with support for HTTPS only, the SCST package is not installed and you can skip this step.

```
   Loaded: loaded (/etc/systemd/system/mediator-scst.service; enabled;
vendor preset: disabled)
   Active: active (running) since Mon 2022-04-18 10:41:47 EDT; 1 weeks 0
days ago
  Process: 286595 ExecStart=/etc/init.d/scst start (code=exited,
status=0/SUCCESS)
 Main PID: 286662 (iscsi-scstd)
    Tasks: 1 (limit: 49473)
   Memory: 1.2M
   CGroup: /system.slice/mediator-scst.service
           └─286662 /usr/local/sbin/iscsi-scstd
```

3. Confirm the ports that are used by ONTAP Mediator:

```
netstat -anlt | grep -E '3260|31784'
```

```
         tcp   0   0 0.0.0.0:31784   0.0.0.0:*       LISTEN

         tcp   0   0 0.0.0.0:3260    0.0.0.0:*       LISTEN

         tcp6  0   0 :::3260         :::*            LISTEN
```

What's next?

After ONTAP Mediator is installed and verified, you can proceed with the following post-installation configuration tasks:

- Use ONTAP Mediator for SnapMirror active sync and MetroCluster IP
- Configure ONTAP Mediator attributes (Optional)

Use ONTAP Mediator

Use ONTAP Mediator for SnapMirror active sync and MetroCluster IP

ONTAP Mediator runs on a Linux host in a third failure domain, separate from your ONTAP clusters, where it provides a persistent, fenced store for high-availability (HA) metadata. ONTAP uses this store, along with the Mediator's node health query and ping-proxy functions, to monitor SnapMirror active sync relationships and MetroCluster IP configurations.

> This documentation refers to the on-premises version of ONTAP Mediator. For information about ONTAP Cloud Mediator, available beginning with ONTAP 9.17.1, see the SnapMirror active sync documentation.

Monitor SnapMirror active sync relationships

SnapMirror active sync uses ONTAP Mediator as a passive witness to the primary and mirror copies in a consistency group. If a network partition occurs or one copy becomes unavailable, SnapMirror active sync queries ONTAP Mediator to determine which copy continues to serve I/O, and discontinues I/O on the other copy.

Beginning with ONTAP 9.8, you can use any version of ONTAP Mediator to monitor a SnapMirror active sync relationship.

> For limits on the number of SnapMirror active sync cluster pairs that a single ONTAP Mediator instance can monitor, see Object limits for ONTAP SnapMirror active sync.

Configure the ONTAP Mediator for SnapMirror active sync

Monitor MetroCluster IP configurations

In a MetroCluster IP configuration, ONTAP Mediator stores state information about the MetroCluster nodes in mailbox LUNs and assists the nodes with a Mediator-assisted automatic unplanned switchover (MAUSO) in the event of a disaster or specific failure scenarios.

Beginning with ONTAP 9.19.1 and ONTAP Mediator 1.12, you can configure an ONTAP Mediator that uses HTTPS for communication to monitor your MetroCluster IP configuration. When you configure ONTAP Mediator with the iSCSI protocol, you can use any version of ONTAP Mediator to monitor a MetroCluster IP configuration.

> For limits on the number of MetroCluster IP configurations that a single ONTAP Mediator instance can monitor, see Use ONTAP Mediator to monitor multiple MetroCluster IP configurations

Configure ONTAP Mediator for a MetroCluster IP configuration

Related information

- ONTAP SnapMirror active sync architecture

- Learn about ONTAP Mediator support for automatic unplanned switchover in a MetroCluster IP configuration

Administer ONTAP Mediator

Upgrade ONTAP Mediator

Upgrade ONTAP Mediator

To upgrade ONTAP Mediator to a newer version without upgrading the host OS, download the installation package for the new version and run the installer on the host where ONTAP Mediator is already installed.

If you need to upgrade the host OS to a new release and reinstall or upgrade ONTAP Mediator as part of that process, see Upgrade the host OS and ONTAP Mediator instead.

Before you begin

- Verify that your host meets the prerequisites described in Prepare to install ONTAP Mediator, including the host, firewall, and OS requirements.
- Download the installation package for the new version. See Download the ONTAP Mediator installation package.
- NetApp recommends verifying the code signature of the new installation package. See Verify the code signature of the ONTAP Mediator installation package.

About this task

Review the following information before upgrading ONTAP Mediator.

- You should upgrade ONTAP Mediator to the latest version. Older versions still work with all ONTAP releases, but newer versions include security patches for third-party components.
- When you upgrade to a new ONTAP Mediator version that includes the SCST package, the installer automatically upgrades to the recommended SCST version unless a higher version is available. For instructions on manually installing a higher SCST version, see Maintain the host OS for ONTAP Mediator. For supported versions, see the SCST support matrix.

- If the upgrade fails, you might need to try a newer version of ONTAP Mediator.
- From June 15, 2025, you can't install or upgrade ONTAP Mediator 1.9 and 1.8 because their code signing certificates have expired. If the upgrade fails, use the ONTAP Mediator 1.9.1 patch version instead.

- Beginning with ONTAP 9.19.1 and ONTAP Mediator 1.12, you have the option to install ONTAP Mediator with iSCSI and HTTPS support, or support for HTTPS only. If you choose to install ONTAP Mediator with support for HTTPS only, the SCST package is not installed.
- If you upgrade from ONTAP Mediator 1.11 or earlier with SCST installed, and you choose the HTTPS only option during the upgrade, the SCST package is uninstalled and the new ONTAP Mediator version does not contain the SCST package.
- If you plan to use ONTAP Mediator to monitor multiple MetroCluster IP configurations, the support depends on your Mediator type, ONTAP version, and OS. Refer to Use ONTAP Mediator to monitor multiple MetroCluster IP configurations for more information.
- Installing ONTAP Mediator with support for HTTPS only is supported for the following use cases:
  - You plan to only use ONTAP Mediator to monitor your SnapMirror active sync clusters.

- You plan to configure ONTAP Mediator with HTTPS to monitor one or more MetroCluster IP configurations.

- ONTAP Mediator must be installed with support for iSCSI and HTTPS in the following scenario:
  - You plan to configure ONTAP Mediator with iSCSI to monitor one or more MetroCluster IP configurations.

- When you run the installer against an existing ONTAP Mediator installation, it detects the current version and prompts you to confirm the upgrade:

```
ONTAP Mediator will be upgraded from version <current-version> to <new-version>.

Do you wish to continue? Y(es)/n(o):
```

- As part of the upgrade, the installer captures a support bundle of your pre-upgrade configuration and migrates the database before upgrading the ONTAP Mediator software.
- For automation use cases, you can use the `./ontap-mediator-1.13.0/ontap-mediator-1.13.0 -y` command to skip the installer prompts.

Steps

1. Before the upgrade, complete the following substep if you are upgrading to ONTAP Mediator 1.13 or later on an ONTAP Mediator instance that pairs with one of the following releases:
   - ONTAP 9.11.1 and earlier
   - ONTAP patch releases earlier than 9.12.1P15, 9.13.1P12, 9.14.1P8, and 9.15.1P2
     a. Add the following line to the `/opt/netapp/lib/ontap_mediator/ontap_mediator/server_config/ontap_mediator.user_config.yaml` file.

        ```
        'generate_unique_ca_name': false
        ```

        This file is preserved across ONTAP Mediator upgrades.

2. Run the installer for the new version and respond to the prompts as required:

   ```
   ./ontap-mediator-1.13.0/ontap-mediator-1.13.0
   ```

   To skip the signature check during the upgrade, use the `./ontap-mediator-1.13.0/ontap-mediator-1.13.0 --skip-code-signature-check` command

   The installer detects the existing installation and prompts you to upgrade, then installs the required packages and migrates your configuration.

   ▼ Example ONTAP Mediator upgrade with HTTPS and iSCSI support (console output)

   ```
   [root@mediator-host ~]# ontap-mediator-1.13.0/ontap-mediator-1.13.0

   + Extracting the ONTAP Mediator installation/upgrade archive
   ```

```
+ Performing the ONTAP Mediator run-time code signature check
   Using openssl from the path: /usr/bin/openssl configured for
CApath:/etc/pki/tls
Error querying OCSP responder
40B70028217F0000:error:1E800080:HTTP
routines:OSSL_HTTP_REQ_CTX_nbio:failed reading
data:crypto/http/http_client.c:596:
40B70028217F0000:error:1E800067:HTTP
routines:OSSL_HTTP_REQ_CTX_exchange:error
receiving:crypto/http/http_client.c:1046:server=http://ocsp.entrust.ne
t:80
   WARNING: The OCSP check failed while attempting to test the Code-
Signature-Check certificate
   Continue without code signature checking (only recommended if
integrity has been established manually)? y(es)/N(o): yes
 SKIPPING: Code signature check, manual override due to lack of OCSP
response
+ Unpacking the ONTAP Mediator installer
+ ONTAP_MEDIATOR_INSTALL_HTTPS_ONLY is set to '0', args are: ''
+ Starting ONTAP Mediator install/update process: previous
installation_mode is ''


ONTAP Mediator can be installed with iSCSI and HTTPS support, or HTTPS
support only.
Do you wish to install ONTAP Mediator with iSCSI and HTTPS support?
Y(es)/n(o): yes
ONTAP Mediator will be installed with mode: ISCSI_HTTPS
Selected install package mode: ISCSI_HTTPS
ONTAP Mediator will be upgraded from version 1.12.1 to 1.13.0.
Do you wish to continue? Y(es)/n(o): yes
Capturing support_bundle to save pre-upgrade configuration. API
Credentials are required.
    Mediator API User Name: mediatoradmin
                 Password: Running plugins, please wait...

Running 1/11: apis...

Creating tar archive...
Support bundle has been generated at
/opt/netapp/data/support_bundles/sdot-r730-0003a-
d4.gdl.englab.netapp.com_1792143997_BASIC.tgz
Testing the DB can be upgraded
Cloning database...
Cloning successful.
Verifying database is up-to-date...
```

```
Database out-of-date.Migrations are necessary:
Current revision: 45ca93e60653, Head revision: 086df085a190
Performing database migrations against clone...
Migrations successful.
Database upgrade operation complete.
+ Adding python312 specific packages
+ Upgrading ONTAP Mediator. (Log: /root/ontap_mediator.FO5UgL/ontap-
mediator-1.13.0/ontap-mediator-1.13.0/upgrade_20261016054617.log)
fapolicyd not installed or no rules.d directory found, skipping
fapolicyd configuration
    ONTAP Mediator rsyslog and logging rotation enabled
+ Upgrade successful. (Moving log to
/opt/netapp/lib/ontap_mediator/log/upgrade_20261016054617.log)\n
+ WARNING: This system supports UEFI
           Secure Boot (SB) is currently enabled on this system.
           The following action need be taken:
           Using the keys in
/opt/netapp/lib/ontap_mediator/ontap_mediator/SCST_mod_keys follow
           instructions in
/opt/netapp/lib/ontap_mediator/ontap_mediator/SCST_mod_keys/README.mod
ule-signing
           to sign the SCST kernel module. Note that reboot will be
needed.
     SCST will not start automatically when Secure Boot is enabled and
not configured properly.

+ Note: ONTAP Mediator generated a self-signed server certificate for
temporary use on
    this host. If the DNS name or IP address for the host is changed,
the certificate
    will no longer be valid. The default certificates should be
replaced with secure
    trusted certificates signed by a known certificate authority prior
to use for production.
    For more information, see /opt/netapp/lib/ontap_mediator/README

+ Note: ONTAP Mediator uses a kernel module compiled specifically for
the current
        OS. Using 'yum update' to upgrade the kernel might cause
service interruption.
    For more information, see /opt/netapp/lib/ontap_mediator/README
[root@mediator-host ~]#
```

▼ **Example of ONTAP Mediator upgrade with support for HTTPS only (console output)**

```
[root@mediator-host ~]# ontap-mediator-1.13.0/ontap-mediator-1.13.0
```

```
--https-only

ONTAP Mediator: Self Extracting Installer

+ Extracting the ONTAP Mediator installation/upgrade archive
+ Performing the ONTAP Mediator run-time code signature check
   Using openssl from the path: /usr/bin/openssl configured for
CApath:/etc/pki/tls
Error querying OCSP responder
40A73D8E847F0000:error:1E800080:HTTP
routines:OSSL_HTTP_REQ_CTX_nbio:failed reading
data:crypto/http/http_client.c:614:
40A73D8E847F0000:error:1E800067:HTTP
routines:OSSL_HTTP_REQ_CTX_exchange:error
receiving:crypto/http/http_client.c:1075:server=http://ocsp.entrust.ne
t:80
   WARNING: The OCSP check failed while attempting to test the Code-
Signature-Check certificate
   Continue without code signature checking (only recommended if
integrity has been established manually)? y(es)/N(o): yes
 SKIPPING: Code signature check, manual override due to lack of OCSP
response
+ Unpacking the ONTAP Mediator installer
+ ONTAP_MEDIATOR_INSTALL_HTTPS_ONLY is set to '1', args are: ''
+ Starting ONTAP Mediator install/update process: previous
installation_mode is 'HTTPS_ONLY'
+ Selected install package mode: HTTPS_ONLY
ONTAP Mediator will be upgraded from version 1.12.1 to 1.13.0.


Do you wish to continue? Y(es)/n(o): yes
Capturing support_bundle to save pre-upgrade configuration. API
Credentials are required.
    Mediator API User Name: mediatoradmin
                 Password: Running plugins, please wait...


Creating tar archive...
Support bundle has been generated at
/opt/netapp/data/support_bundles/sdot-r730-0003a-
d4.gdl.englab.netapp.com_1792151055_BASIC.tgz
Testing the DB can be upgraded
Cloning database...
Cloning successful.
Verifying database is up-to-date...
Database out-of-date.Migrations are necessary:
```

```
Current revision: 45ca93e60653, Head revision: 086df085a190
Performing database migrations against clone...
Migrations successful.
Database upgrade operation complete.
+ Adding python312 specific packages
+ Upgrading ONTAP Mediator. (Log: /root/ontap_mediator.qmrGsH/ontap-
mediator-1.13.0/ontap-mediator-1.13.0/upgrade_20261016074342.log)
fapolicyd not installed or no rules.d directory found, skipping
fapolicyd configuration
    ONTAP Mediator rsyslog and logging rotation enabled
+ Upgrade successful. (Moving log to
/opt/netapp/lib/ontap_mediator/log/upgrade_20261016074342.log)\n
+ WARNING: This system supports UEFI
+ Note: Not signing the SCST modules since they were not installed by
this installation.
+ Note: SCST will not start automatically if Secure Boot is enabled
and not configured properly.

+ Note: ONTAP Mediator generated a self-signed server certificate for
temporary use on
    this host. If the DNS name or IP address for the host is changed,
the certificate
    will no longer be valid. The default certificates should be
replaced with secure
    trusted certificates signed by a known certificate authority prior
to use for production.
    For more information, see /opt/netapp/lib/ontap_mediator/README

+ Note: ONTAP Mediator uses a kernel module compiled specifically for
the current
        OS. Using 'yum update' to upgrade the kernel might cause
service interruption.
    For more information, see /opt/netapp/lib/ontap_mediator/README
[root@mediator-host ~]#
```

Upgrade the host OS and reinstall or upgrade ONTAP Mediator

To upgrade the host OS for ONTAP Mediator with SCST installed to a later version, you must uninstall ONTAP Mediator. After the OS upgrade, you can reinstall your current version of ONTAP Mediator or upgrade to a newer version.

About this task

- This task applies to ONTAP Mediator versions with SCST installed. If you installed ONTAP Mediator 1.12 or later with support for HTTPS only, the SCST package is not installed and you can upgrade the host OS without affecting ONTAP Mediator. Refer to the host OS vendor's documentation for OS upgrade instructions.

- If you only need to update the host OS kernel to a supported version without upgrading to a new OS release, see Maintain the host OS for ONTAP Mediator instead.
- Before upgrading the host OS for ONTAP Mediator with the leapp-upgrade tool, uninstall ONTAP Mediator. The tool checks for new RPM versions in registered repositories.
- The ONTAP Mediator installer installs an .rpm file, which the leapp-upgrade tool includes in the search. Because the installer unpacks the file instead of downloading it from a registered repository, the tool cannot find an upgrade. You must use the leapp-upgrade tool to uninstall the package.

Steps

1. Back up the log files:

```
[rootmediator-host ~]# tar -czf ontap_mediator_file_backup.tgz -C
/opt/netapp/lib/ontap_mediator ./log
./ontap_mediator/server_config/ontap_mediator.user_config.yaml
[rootmediator-host ~]# tar -tf ontap_mediator_file_backup.tgz
./log/
./log/ontap_mediator.log
./log/scstadmin.log
./log/ontap_mediator_stdout.log
./log/ontap_mediator_requests.log
./log/install_20230419134611.log
./log/scst.log
./log/ontap_mediator_syslog.log
./ontap_mediator/server_config/ontap_mediator.user_config.yaml
[rootmediator-host ~]#
```

2. Perform upgrade with leapp-upgrade tool:

```
[rootmediator-host ~]# leapp preupgrade --target 9.8
  ..<snip upgrade checks>..
  ..<fix issues found>..
[rootmediator-host ~]# leapp upgrade --target 9.8
  ..<snip upgrade>..
[rootmediator-host ~]# cat /etc/os-release | head -2
NAME="Red Hat Enterprise Linux"
VERSION="8.4 (Ootpa)"
[rootmediator-host ~]#
```

3. Reinstall ONTAP Mediator:

> Perform the rest of the steps immediately after reinstalling ONTAP Mediator to prevent a loss of log files.

```
[rootmediator-host ~]# ontap-mediator-1.12.1/ontap-mediator-1.12.1
```

```
ONTAP Mediator: Self Extracting Installer

  ..<snip installation>..
[rootmediator-host ~]#
```

4. Stop ontap_mediator:

```
[rootmediator-host ~]# systemctl stop ontap_mediator
[rootmediator-host ~]#
```

5. Replace the log files:

```
[rootmediator-host ~]# tar -xf ontap_mediator_file_backup.tgz -C
/opt/netapp/lib/ontap_mediator
[rootmediator-host ~]#
```

6. Start ontap_mediator:

```
[rootmediator-host ~]# systemctl start ontap_mediator
[rootmediator-host ~]#
```

7. Reconnect all ONTAP clusters to the upgraded ONTAP Mediator:

MetroCluster over IP

```
siteA::> metrocluster configuration-settings mediator show
Mediator IP     Port    Node                    Configuration
Connection
                                                Status        Status
--------------- ------- ---------------------- -------------
-----------
172.31.40.122
                31784   siteA-node2             true          false
                        siteA-node1             true          false
                        siteB-node2             true          false
                        siteB-node2             true          false
siteA::> metrocluster configuration-settings mediator remove
Removing the mediator and disabling Automatic Unplanned Switchover. It
may take a few minutes to complete.
Please enter the username for the mediator: mediatoradmin
Please enter the password for the mediator:
Confirm the mediator password:
Automatic Unplanned Switchover is disabled for all nodes...
```

```
Removing mediator mailboxes...
Successfully removed the mediator.

siteA::> metrocluster configuration-settings mediator add -mediator
-address 172.31.40.122
Adding the mediator and enabling Automatic Unplanned Switchover. It may
take a few minutes to complete.
Please enter the username for the mediator: mediatoradmin
Please enter the password for the mediator:
Confirm the mediator password:
Successfully added the mediator.

siteA::> metrocluster configuration-settings mediator show
Mediator IP     Port    Node                    Configuration
Connection
                                                Status        Status
--------------- ------- ----------------------- -------------
-----------
172.31.40.122
                31784   siteA-node2             true          true
                        siteA-node1             true          true
                        siteB-node2             true          true
                        siteB-node2             true          true
siteA::>
```

SnapMirror active sync

For SnapMirror active sync, you do not need to reinstall TLS certificates stored outside /opt/netapp. Back up and restore the certificates stored in /opt/netapp.

```
peer1::> snapmirror mediator show
Mediator Address Peer Cluster     Connection Status Quorum Status
---------------- ---------------- ----------------- -------------
172.31.49.237    peer2            unreachable       true

peer1::> snapmirror mediator remove -mediator-address 172.31.49.237
-peer-cluster peer2

Info: [Job 39] 'mediator remove' job queued

peer1::> job show -id 39
                            Owning
Job ID Name                 Vserver    Node           State
------ -------------------- ---------- -------------- ----------
39     mediator remove      peer1      peer1-node1    Success
     Description: Removing entry in mediator
```

```
peer1::> security certificate show -common-name ONTAPMediatorCA
Vserver    Serial Number   Certificate Name                       Type
---------- --------------- -------------------------------------
------------
peer1
        4A790360081F41145E14C5D7CE721DC6C210007F
                        ONTAPMediatorCA                           server-
ca
    Certificate Authority: ONTAP Mediator CA
        Expiration Date: Mon Apr 17 10:27:54 2073

peer1::> security certificate delete -common-name ONTAPMediatorCA *
1 entry was deleted.

 peer1::> security certificate install -type server-ca -vserver peer1

Please enter Certificate: Press <Enter> when done
  ..<snip ONTAP Mediator CA public key>..

You should keep a copy of the CA-signed digital certificate for future
reference.

The installed certificate's CA and serial number for reference:
CA: ONTAP Mediator CA
serial: 44786524464C5113D5EC966779D3002135EA4254

The certificate's generated name for reference: ONTAPMediatorCA

peer2::> security certificate delete -common-name ONTAPMediatorCA *
1 entry was deleted.

peer2::> security certificate install -type server-ca -vserver peer2

 Please enter Certificate: Press <Enter> when done
..<snip ONTAP Mediator CA public key>..


You should keep a copy of the CA-signed digital certificate for future
reference.

The installed certificate's CA and serial number for reference:
CA: ONTAP Mediator CA
serial: 44786524464C5113D5EC966779D3002135EA4254

The certificate's generated name for reference: ONTAPMediatorCA

peer1::> snapmirror mediator add -mediator-address 172.31.49.237 -peer
```

```
-cluster peer2 -username mediatoradmin

Notice: Enter the mediator password.

Enter the password:
Enter the password again:

Info: [Job: 43] 'mediator add' job queued

peer1::> job show -id 43
                            Owning
Job ID Name                 Vserver    Node           State
------ -------------------- ---------- -------------- ----------
43     mediator add         peer1      peer1-node2    Success
    Description: Creating a mediator entry

peer1::> snapmirror mediator show
Mediator Address Peer Cluster     Connection Status Quorum Status
---------------- ---------------- ----------------- -------------
172.31.49.237    peer2            connected         true

peer1::>
```

Related information

- security certificate delete
- security certificate install
- security certificate show
- snapmirror mediator add
- snapmirror mediator remove
- storage iscsi-initiator show

Configure ONTAP Mediator attributes

You can configure ONTAP Mediator host-side attributes, including certificate-related settings and configuration overrides.

ONTAP Mediator configurable settings

ONTAP Mediator supports several configurable settings. The default values for these settings are provided in a `ontap_mediator.config.yaml` read-only file. The file is located at:

`/opt/netapp/lib/ontap_mediator/ontap_mediator/server_config/ontap_mediator.user_config.yaml`

All values that are placed in the `ontap_mediator.user_config.yaml` will override the default values and

be maintained across all ONTAP Mediator upgrades.

After you modify `ontap_mediator.user_config.yaml`, restart ONTAP Mediator:

```
systemctl restart ontap_mediator
```

The ONTAP Mediator attributes described in this section can be modified if required.

> Other default values in the `ontap_mediator.config.yaml` should not be changed because modified values are not maintained during ONTAP Mediator upgrades.

You modify ONTAP Mediator attributes by copying the required variables to the `ontap_mediator.user_config.yaml` file to override the default settings.

Install third-party SSL certificates

You can replace the default self-signed certificates with third-party SSL certificates by modifying the ONTAP Mediator configuration files.

To replace the default self-signed certificates with third-party SSL certificates, you modify certain attributes in the following files:

- `/opt/netapp/lib/ontap_mediator/ontap_mediator/server_config/ontap_mediator.config.yaml`
- `/opt/netapp/lib/ontap_mediator/uwsgi/ontap_mediator.ini`

The variables in these files are used to control the certificate files used by ONTAP Mediator.

ONTAP Mediator 1.9 and later

The default variables listed in the following table are included in the `/opt/netapp/lib/ontap_mediator/ontap_mediator/server_config/ontap_mediator.config.yaml` file.

Variable	Path
`cert_path`	`/opt/netapp/lib/ontap_mediator/ontap_mediator/server_config/ontap_mediator_server.crt`
`key_path`	`/opt/netapp/lib/ontap_mediator/ontap_mediator/server_config/ontap_mediator_server.key`
`ca_cert_path`	`/opt/netapp/lib/ontap_mediator/ontap_mediator/server_config/intermediate.crt`
`ca_key_path`	`/opt/netapp/lib/ontap_mediator/ontap_mediator/server_config/intermediate.key`
`ca_serial_path`	`/opt/netapp/lib/ontap_mediator/ontap_mediator/server_config/intermediate.srl`

Variable	Path
cert_valid_days	1095
x509_passin_pwd	pass:ontap

- `cert_valid_days` is used to set the expiration of client certificates. The maximum value is three years (1095 days).
- `x509_passin_pwd` is the passphrase for the signed client certificate.

The default variables listed in the following table are included in the `/opt/netapp/lib/ontap_mediator/uwsgi/ontap_mediator.ini` file.

Variable	Path
mediator_cert	/opt/netapp/lib/ontap_mediator/ontap_mediator/server_config/ontap_mediator_server.crt
mediator_key	/opt/netapp/lib/ontap_mediator/ontap_mediator/server_config/ontap_mediator_server.key
ca_cert_path	/opt/netapp/lib/ontap_mediator/ontap_mediator/server_config/intermediate.crt

ONTAP Mediator 1.8 and earlier

The default variables listed in the following table are included in the `/opt/netapp/lib/ontap_mediator/ontap_mediator/server_config/ontap_mediator.config.yaml` file.

Variable	Path
cert_path	/opt/netapp/lib/ontap_mediator/ontap_mediator/server_config/ontap_mediator_server.crt
key_path	/opt/netapp/lib/ontap_mediator/ontap_mediator/server_config/ontap_mediator_server.key
ca_cert_path	/opt/netapp/lib/ontap_mediator/ontap_mediator/server_config/ca.crt
ca_key_path	/opt/netapp/lib/ontap_mediator/ontap_mediator/server_config/ca.key
ca_serial_path	/opt/netapp/lib/ontap_mediator/ontap_mediator/server_config/ca.srl
cert_valid_days	1095
x509_passin_pwd	pass:ontap

- `cert_valid_days` is used to set the expiration of client certificates. The maximum value is three years (1095 days).

- `x509_passin_pwd` is the passphrase for the signed client certificate.

The default variables listed in the following table are included in the `/opt/netapp/lib/ontap_mediator/uwsgi/ontap_mediator.ini` file.

Variable	Path
`mediator_cert`	`/opt/netapp/lib/ontap_mediator/ontap_mediator/server_config/ontap_mediator_server.crt`
`mediator_key`	`/opt/netapp/lib/ontap_mediator/ontap_mediator/server_config/ontap_mediator_server.key`
`ca_cert_path`	`/opt/netapp/lib/ontap_mediator/ontap_mediator/server_config/ca.crt`

If you modify these attributes, restart ONTAP Mediator to apply the changes. For detailed instructions on how to replace default certificates with third-party certificates, refer to Replace self-signed certificates with trusted third-party certificates.

Configure password attack protection

The following settings provide protection against brute-force password guessing attacks.

To enable the feature, set a value for the `window_seconds` and the `retry_limit`.

Examples:

- Provide a 5-minute window for guesses, and then reset the count to zero failures:

  ```
  authentication_lock_window_seconds: 300
  ```

- Lock the account if five failures occur within the window timeframe:

  ```
  authentication_retry_limit: 5
  ```

- Reduce the impact of brute-force password guessing attacks by setting a delay that occurs prior to rejecting each attempt, which slows the attacks.

  ```
  authentication_failure_delay_seconds: 5
  ```

```
authentication_failure_delay_seconds: 0   # seconds (float) to delay
failed auth attempts prior to response, 0 = no delay
authentication_lock_window_seconds: null  # seconds (int) since the
oldest failure before resetting the retry counter, null = no window
authentication_retry_limit: null          # number of retries to allow
before locking API access, null = unlimited
```

Configure password complexity rules

The following fields control the password complexity rules of the ONTAP Mediator API user account.

```
password_min_length: 8

password_max_length: 64

password_uppercase_chars: 0    # min. uppercase characters

password_lowercase_chars: 1    # min. lowercase character

password_special_chars: 1      # min. non-letter, non-digit

password_nonletter_chars: 2    # min. non-letter characters (digits,
specials, anything)
```

Configure ONTAP Mediator disk and log space settings, including the free-space threshold and reserved log space.

Configure the free-space threshold

There are settings that control the required free space on the `/opt/netapp/lib/ontap_mediator` disk.

If the space is lower than the set threshold, the service issues a warning event.

```
low_space_threshold_mib: 10
```

Configure reserved log space

The `RESERVE_LOG_SPACE` feature is controlled by specific settings. By default, the ONTAP Mediator installation creates a separate disk space for the logs. The installer creates a new fixed-size file with a total of 700MB of disk space to be used explicitly for ONTAP Mediator logging.

To disable this feature and use the default disk space, perform the following steps:

1. Change the value of RESERVE_LOG_SPACE from 1 to 0 in the following file:

   `/opt/netapp/lib/ontap_mediator/tools/mediator_env`

   a. Run the following command:

      ```
      cat /opt/netapp/lib/ontap_mediator/tools/mediator_env | grep
      "RESERVE_LOG_SPACE"
      ```

```
RESERVE_LOG_SPACE=0
```

b. Restart ONTAP Mediator:

```
systemctl restart ontap_mediator
```

To re-enable the feature, change the value from 0 to 1 and restart the Mediator.

Toggling between disk spaces does not purge existing logs. All previous logs are backed up and then moved to the current disk space after toggling and restarting the Mediator.

= Change the ONTAP Mediator username and password
:icons: font
:relative_path: ./administer/
:imagesdir: /tmp/d20261109-3041152-j9lm3i/source/./administer/../media/

Change the ONTAP Mediator username or password from the Linux host where ONTAP Mediator is installed.

== Change the username

You can change the username using the following procedure.

About this task

Perform this task on the Linux host where you installed ONTAP Mediator.

If you are unable to reach this command, you might need to run the command using the full path as shown in the following example:

```
/usr/local/bin/mediator_change_user
```

Steps

Change the username by choosing one of the following options:

- **Option (a)**: Run the command `mediator_change_user` and respond to the prompts as shown in the following example:

```
[root@mediator-host ~]# mediator_change_user
Modify the Mediator API username by entering the following values:
    Mediator API User Name: mediatoradmin
                  Password:
New Mediator API User Name: mediator
The account username has been modified successfully.
[root@mediator-host ~]#
```

- **Option (b)**: Run the following command:

```
MEDIATOR_USERNAME=mediator MEDIATOR_PASSWORD=mediator2
MEDIATOR_NEW_USERNAME=mediatoradmin mediator_change_user
```

```
 [root@mediator-host ~]# MEDIATOR_USERNAME=mediator
MEDIATOR_PASSWORD='mediator2' MEDIATOR_NEW_USERNAME=mediatoradmin
mediator_change_user
 The account username has been modified successfully.
 [root@mediator-host ~]#
```

== Change the password

You can change the password using the following procedure.

About this task

Perform this task on the Linux host where you installed ONTAP Mediator.

If you are unable to reach this command, you might need to run the command using the full path as shown in the following example:

```
/usr/local/bin/mediator_change_password
```

Steps

Change the password by choosing one of the following options:

- **Option (a)**: Run the `mediator_change_password` command and respond to the prompts as shown in the following example:

```
[root@mediator-host ~]# mediator_change_password
Change the Mediator API password by entering the following values:
   Mediator API User Name: mediatoradmin
             Old Password:
             New Password:
         Confirm Password:
The password has been updated successfully.
[root@mediator-host ~]#
```

- **Option (b)**: Run the following command:

```
MEDIATOR_USERNAME=mediatoradmin MEDIATOR_PASSWORD=mediator1
MEDIATOR_NEW_PASSWORD=mediator2 mediator_change_password
```

The example shows that the password is changed from "mediator1" to "mediator2".

```
 [root@mediator-host ~]# MEDIATOR_USERNAME=mediatoradmin
MEDIATOR_PASSWORD=mediator1 MEDIATOR_NEW_PASSWORD=mediator2
mediator_change_password
 The password has been updated successfully.
```

```
[root@mediator-host ~]#
```

= Start and stop ONTAP Mediator
:icons: font
:relative_path: ./administer/
:imagesdir: /tmp/d20261109-3041152-j9lm3i/source/./administer/../media/

Stop or restart ONTAP Mediator and SCST when you need to maintain the host or re-enable the service.

== Stop ONTAP Mediator

To stop ONTAP Mediator, perform the following steps:

Steps

1. Stop ONTAP Mediator:

```
systemctl stop ontap_mediator
```

2. Stop SCST:

   Skip this step if you installed ONTAP Mediator with support for HTTPS only.

```
systemctl stop mediator-scst
```

3. Disable ONTAP Mediator and SCST:

```
systemctl disable ontap_mediator mediator-scst
```

== Start ONTAP Mediator

To re-enable ONTAP Mediator, perform the following steps:

Steps

1. Enable ONTAP Mediator and SCST:

```
systemctl enable ontap_mediator mediator-scst
```

2. Start SCST:

   Skip this step if you installed ONTAP Mediator with support for HTTPS only.

```
systemctl start mediator-scst
```

3. Start ONTAP Mediator:

```
systemctl start ontap_mediator
```

= Verify ONTAP Mediator health
:icons: font
:relative_path: ./administer/
:imagesdir: /tmp/d20261109-3041152-j9lm3i/source/./administer/../media/

Verify that ONTAP Mediator is running normally.

== Verify the service status

Steps

1. View the status of ONTAP Mediator:
   a. Run the following command:

```
systemctl status ontap_mediator
```

```
[root@mediator_host ~]# systemctl status ontap_mediator

 ontap_mediator.service - ONTAP Mediator
Loaded: loaded (/etc/systemd/system/ontap_mediator.service; enabled;
vendor preset: disabled)
Active: active (running) since Mon 2022-04-18 10:41:49 EDT; 1 weeks 0
days ago
Process: 286710 ExecStop=/bin/kill -s INT $MAINPID (code=exited,
status=0/SUCCESS)
Main PID: 286712 (uwsgi)
Status: "uWSGI is ready"
Tasks: 3 (limit: 49473)
Memory: 139.2M
CGroup: /system.slice/ontap_mediator.service
      ├─286712 /opt/netapp/lib/ontap_mediator/pyenv/bin/uwsgi --ini
/opt/netapp/lib/ontap_mediator/uwsgi/ontap_mediator.ini
      ├─286716 /opt/netapp/lib/ontap_mediator/pyenv/bin/uwsgi --ini
/opt/netapp/lib/ontap_mediator/uwsgi/ontap_mediator.ini
      └─286717 /opt/netapp/lib/ontap_mediator/pyenv/bin/uwsgi --ini
/opt/netapp/lib/ontap_mediator/uwsgi/ontap_mediator.ini

[root@mediator_host ~]#
```

   b. View the status of the SCST service:

If you installed ONTAP Mediator 1.12 or later with support for HTTPS only, the SCST package is not installed and you can skip this task.

```
systemctl status mediator-scst
```

```
[root@mediator_host ~]# systemctl status mediator-scst
   Loaded: loaded (/etc/systemd/system/mediator-scst.service;
enabled; vendor preset: disabled)
   Active: active (running) since Mon 2022-04-18 10:41:47 EDT; 1
weeks 0 days ago
  Process: 286595 ExecStart=/etc/init.d/scst start (code=exited,
status=0/SUCCESS)
 Main PID: 286662 (iscsi-scstd)
    Tasks: 1 (limit: 49473)
   Memory: 1.2M
   CGroup: /system.slice/mediator-scst.service
           └─286662 /usr/local/sbin/iscsi-scstd

[root@mediator_host ~]#
```

== Confirm the listening ports

1. Confirm the ports that are used by ONTAP Mediator:

For an HTTPS-only installation, expect port 31784 only. The port 3260 listeners in the example below apply when SCST is installed.

```
netstat -anlt | grep -E '3260|31784'
```

```
[root@scspr1905507001 ~]# netstat -anlt | grep -E '3260|31784'

         tcp   0   0 0.0.0.0:31784   0.0.0.0:*       LISTEN

         tcp   0   0 0.0.0.0:3260    0.0.0.0:*       LISTEN

         tcp6  0   0 :::3260         :::*            LISTEN
```

= Manage certificates

= Regenerate a temporary self-signed certificate for the ONTAP Mediator host
:icons: font
:relative_path: ./administer/
:imagesdir: /tmp/d20261109-3041152-j9lm3i/source/./administer/../media/

Regenerate a temporary self-signed certificate for the current host when a hostname or IP address change invalidates the existing certificate.

This procedure is only supported on systems running ONTAP Mediator 1.7 or later.

About this task

- Perform this task on the Linux host where you installed ONTAP Mediator.
- You can perform this task only if the generated self-signed certificates have become obsolete due to changes to the hostname or IP address of the host after installing ONTAP Mediator.
- After the temporary self-signed certificate has been replaced by a trusted third-party certificate, you do *not* use this task to regenerate a certificate. The absence of a self-signed certificate will cause this procedure to fail.

Step

To regenerate a new temporary self-signed certificate for the current host, perform the following step:

1. Restart ONTAP Mediator:

   ```
   ./make_self_signed_certs.sh overwrite
   ```

   ```
   [root@xyz000123456 ~]# cd
   /opt/netapp/lib/ontap_mediator/ontap_mediator/server_config
   [root@xyz000123456 server_config]# ./make_self_signed_certs.sh overwrite

   Adding Subject Alternative Names to the self-signed server certificate
   #
   # OpenSSL example configuration file.
   Generating self-signed certificates
   Generating RSA private key, 4096 bit long modulus (2 primes)
   ..............................................................................
   ..............................................................................
   ..................++++
   ..........................................................++++
   e is 65537 (0x010001)
   Generating a RSA private key
   ...............................................++++
   ..............................................................................
   ..............................................................................+++
   +
   writing new private key to 'ontap_mediator_server.key'
   -----
   Signature ok
   subject=C = US, ST = California, L = San Jose, O = "NetApp, Inc.", OU =
   ONTAP Core Software, CN = ONTAP Mediator, emailAddress =
   support@netapp.com
   Getting CA Private Key
   ```

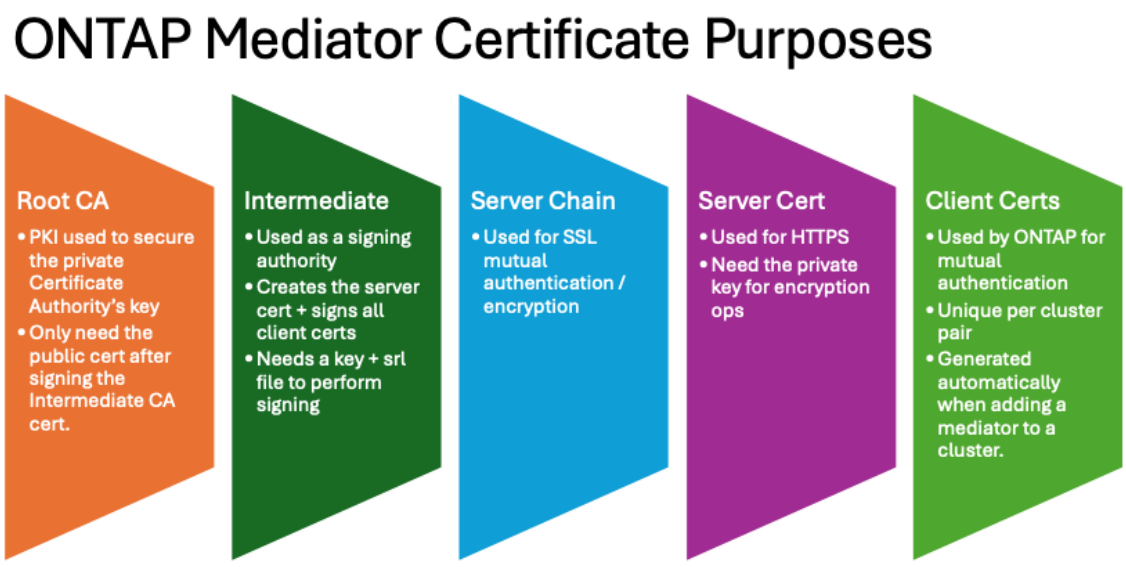
= Replace ONTAP Mediator self-signed certificates with trusted third-party certificates
:icons: font
:relative_path: ./administer/
:imagesdir: /tmp/d20261109-3041152-j9lm3i/source/./administer/../media/

If your environment supports it, replace self-signed certificates with trusted certificates issued by a certificate authority.

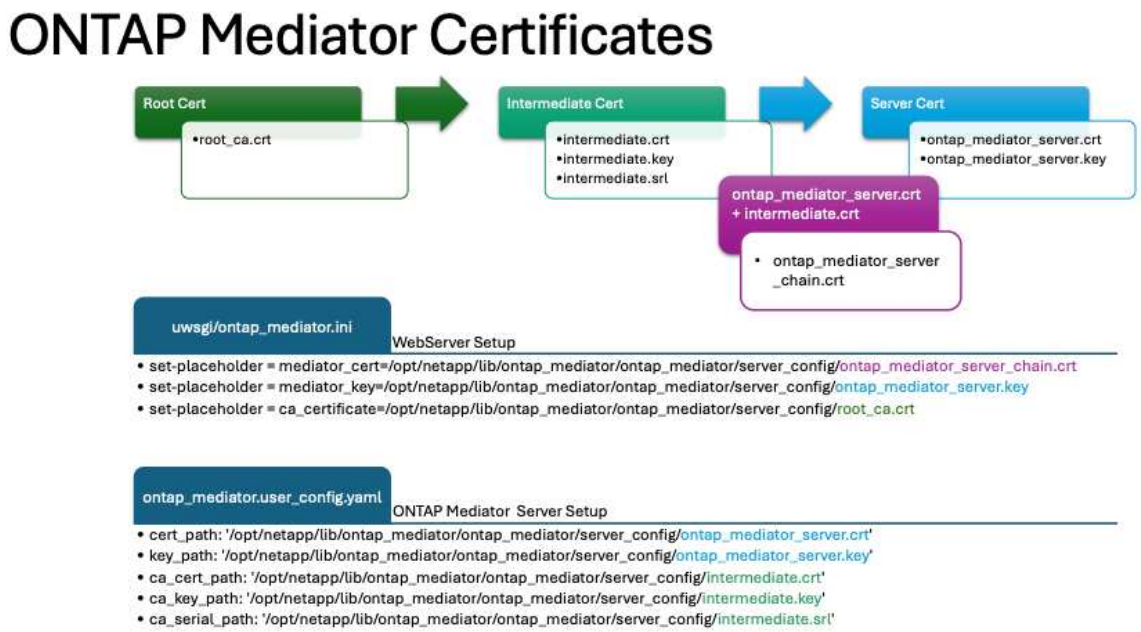
- Third-party certificates are only supported beginning with ONTAP 9.16.1 and in some earlier ONTAP patch releases. See NetApp Bugs Online Bug ID CONTAP-243278.
- Third-party certificates are only supported on systems running ONTAP Mediator 1.7 or later.

About this task

- Perform this task on the Linux host where you installed ONTAP Mediator.
- You can perform this task if the generated self-signed certificates need to be replaced by certificates obtained from a trusted subordinate certificate authority (CA). To accomplish this, you should have access to a trusted public-key infrastructure (PKI) authority.
- The following image shows the purposes of each ONTAP Mediator certificate.



- The following image shows configuration for the web server setup and ONTAP Mediator setup.



== Step 1: Obtain a certificate from a third-party issuing a CA certificate

You can obtain a certificate from a PKI authority using the following procedure.

The following example demonstrates replacing the self-signed certificate actors with the third-party certificate actors located at `/opt/netapp/lib/ontap_mediator/ontap_mediator/server_config/`.

> The example illustrates the necessary criteria for the certificates required for ONTAP Mediator. You can obtain the certificates from a PKI authority in a way that might be different to this procedure. Adjust the procedure according to your business need.

1. Create a private key `intermediate.key` and a configuration file `openssl_ca.cnf` that will be consumed by the PKI authority to generate a certificate.
   a. Generate the private key `intermediate.key`:

      Example

      ```
      openssl genrsa -aes256 -out intermediate.key 4096
      ```

   b. The configuration file `openssl_ca.cnf` (located at `/opt/netapp/lib/ontap_mediator/ontap_mediator/server_config/openssl_ca.cnf`) defines the properties that the generated certificate must have.
2. Use the private key and configuration file to create a certificate signing request `intermediate.csr`:

   Example:

   ```
   openssl req -key <private_key_name>.key -new -out <certificate_csr_name>.csr
   -config <config_file_name>.cnf
   ```

```
[root@mediator_host server_config]# openssl req -key intermediate.key
-new -config openssl_ca.cnf -out intermediate.csr
Enter pass phrase for intermediate.key:
[root@mediator_host server_config]# cat intermediate.csr
-----BEGIN CERTIFICATE REQUEST-----
<certificate_value>
-----END CERTIFICATE REQUEST-----
```

3. Send the certificate signing request `intermediate.csr` to a PKI authority for their signature.

   The PKI authority verifies the request and signs the `.csr`, generating the certificate `intermediate.crt`. Additionally, you need to obtain the `root_ca.crt` certificate that signed the `intermediate.crt` certificate from the PKI authority.

   > For SnapMirror active sync clusters and MetroCluster IP configurations using ONTAP Mediator configured with HTTPS, you must add the `intermediate.crt` and `root_ca.crt` certificates to an ONTAP cluster. See Configure ONTAP Mediator and clusters for SnapMirror active sync and Configure the ONTAP Mediator for a MetroCluster IP configuration.

1. Create a private key `ca.key` and a configuration file `openssl_ca.cnf` that will be consumed by the PKI authority to generate a certificate.

   a. Generate the private key `ca.key`:

      Example:

      ```
      openssl genrsa -aes256 -out ca.key 4096
      ```

   b. The configuration file `openssl_ca.cnf` (located at `/opt/netapp/lib/ontap_mediator/ontap_mediator/server_config/openssl_ca.cnf`) defines the properties that the generated certificate must have.

2. Use the private key and configuration file to create a certificate signing request `ca.csr`:

   Example:

   ```
   openssl req -key <private_key_name>.key -new -out <certificate_csr_name>.csr
   -config <config_file_name>.cnf
   ```

   ```
   [root@mediator_host server_config]# openssl req -key ca.key -new -config
   openssl_ca.cnf -out ca.csr
   Enter pass phrase for ca.key:
   [root@mediator_host server_config]# cat ca.csr
   -----BEGIN CERTIFICATE REQUEST-----
   <certificate_value>
   -----END CERTIFICATE REQUEST-----
   ```

3. Send the certificate signing request `ca.csr` to a PKI authority for their signature.

The PKI authority verifies the request and signs the `.csr`, generating the certificate `ca.crt`. Additionally, you need to obtain the `root_ca.crt` certificate that signed the `ca.crt` certificate from the PKI authority.

For SnapMirror active sync clusters, you must add the `ca.crt` and `root_ca.crt` certificates to an ONTAP cluster. See Configure ONTAP Mediator and clusters for SnapMirror active sync.

== Step 2: Generate a server certificate by signing with a third-party CA certification

A server certificate must be signed by the private key `intermediate.key` and the third-party certificate `intermediate.crt`. Additionally, the configuration file `/opt/netapp/lib/ontap_mediator/ontap_mediator/server_config/openssl_server.cnf` contains certain attributes that specify the properties required for server certificates issued by OpenSSL.

The following commands can generate a server certificate.

Steps

1. To generate a server certificate signing request (CSR), run the following command from the `/opt/netapp/lib/ontap_mediator/ontap_mediator/server_config` folder:

   ```
   openssl req -config openssl_server.cnf -extensions v3_req -nodes -newkey
   rsa:4096 -sha512 -keyout ontap_mediator_server.key -out
   ontap_mediator_server.csr
   ```

2. To generate a server certificate from the CSR, run the following command from the `/opt/netapp/lib/ontap_mediator/ontap_mediator/server_config` folder:

   These files were obtained from a PKI authority. If you are using a different certificate name, replace `intermediate.crt` and `intermediate.key` with the relevant file names.

   ```
   openssl x509 -extfile openssl_server.cnf -extensions v3_req -CA
   intermediate.crt -CAkey intermediate.key -CAcreateserial -sha512 -days 1095
   -req -in ontap_mediator_server.csr -out ontap_mediator_server.crt
   ```

   - The `-CAcreateserial` option is used to generate the `intermediate.srl` files.

A server certificate must be signed by the private key `ca.key` and the third-party certificate `ca.crt`. Additionally, the configuration file `/opt/netapp/lib/ontap_mediator/ontap_mediator/server_config/openssl_server.cnf` contains certain attributes that specify the properties required for server certificates issued by OpenSSL.

The following commands can generate a server certificate.

Steps

1. To generate a server certificate signing request (CSR), run the following command from the `/opt/netapp/lib/ontap_mediator/ontap_mediator/server_config` folder:

   ```
   openssl req -config openssl_server.cnf -extensions v3_req -nodes -newkey
   rsa:4096 -sha512 -keyout ontap_mediator_server.key -out
   ontap_mediator_server.csr
   ```

2. To generate a server certificate from the CSR, run the following command from the `/opt/netapp/lib/ontap_mediator/ontap_mediator/server_config` folder:

   These files were obtained from a PKI authority. If you are using a different certificate name, replace `ca.crt` and `ca.key` with the relevant file names.

   ```
   openssl x509 -extfile openssl_server.cnf -extensions v3_req -CA ca.crt -CAkey
   ca.key -CAcreateserial -sha512 -days 1095 -req -in ontap_mediator_server.csr
   -out ontap_mediator_server.crt
   ```

   - The `-CAcreateserial` option is used to generate the `ca.srl` files.

== Step 3: Replace new third-party CA certificate and server certificate in ONTAP Mediator configuration

The certificate configuration is supplied to ONTAP Mediator in the configuration file located at `/opt/netapp/lib/ontap_mediator/ontap_mediator/server_config/ontap_mediator.config.yaml`. The file includes the following attributes:

```
cert_path:
'/opt/netapp/lib/ontap_mediator/ontap_mediator/server_config/ontap_mediato
r_server.crt'
key_path:
'/opt/netapp/lib/ontap_mediator/ontap_mediator/server_config/ontap_mediato
r_server.key'
ca_cert_path:
'/opt/netapp/lib/ontap_mediator/ontap_mediator/server_config/intermediate.
crt'
ca_key_path:
'/opt/netapp/lib/ontap_mediator/ontap_mediator/server_config/intermediate.
key'
ca_serial_path:
'/opt/netapp/lib/ontap_mediator/ontap_mediator/server_config/intermediate.
srl'
```

- `cert_path` and `key_path` are server certificate variables.
- `ca_cert_path`, `ca_key_path`, and `ca_serial_path` are CA certificate variables.

Steps

1. Replace all `intermediate.*` files with the third-party certificates.
2. Create a certificate chain from the `intermediate.crt` and `ontap_mediator_server.crt` certificates:

   ```
   cat ontap_mediator_server.crt intermediate.crt >
   ontap_mediator_server_chain.crt
   ```

3. Update the `/opt/netapp/lib/ontap_mediator/uvicorn/config.json` file.

   Update the values of `ssl_keyfile`, `ssl_certfile`, and `ssl_ca_certs`:

   ```
   ssl_keyfile:
   /opt/netapp/lib/ontap_mediator/ontap_mediator/server_config/ontap_mediator_ser
   ver.key

   ssl_certfile:
   /opt/netapp/lib/ontap_mediator/ontap_mediator/server_config/ontap_mediator_ser
   ver_chain.crt

   ssl_ca_certs:
   /opt/netapp/lib/ontap_mediator/ontap_mediator/server_config/root_ca.crt
   ```

   - The `ssl_keyfile` value is the key path in the `ontap_mediator_server.crt` file, which is `ontap_mediator_server.key`.
   - The `ssl_certfile` value is the path of the `ontap_mediator_server_chain.crt` file.
   - The `ssl_ca_certs` value is the path of the `root_ca.crt` file.

4. Verify that the following attributes of the newly generated certificates are set correctly:
   - Linux Group Owner: `netapp:netapp`
   - Linux permissions: `600`

5. Restart ONTAP Mediator:

   ```
   systemctl restart ontap_mediator
   ```

The certificate configuration is supplied to ONTAP Mediator in the configuration file located at `/opt/netapp/lib/ontap_mediator/ontap_mediator/server_config/ontap_mediator.config.yaml`. The file includes the following attributes:

```
cert_path:
'/opt/netapp/lib/ontap_mediator/ontap_mediator/server_config/ontap_mediato
r_server.crt'
key_path:
'/opt/netapp/lib/ontap_mediator/ontap_mediator/server_config/ontap_mediato
r_server.key'
ca_cert_path:
'/opt/netapp/lib/ontap_mediator/ontap_mediator/server_config/intermediate.
crt'
ca_key_path:
'/opt/netapp/lib/ontap_mediator/ontap_mediator/server_config/intermediate.
key'
ca_serial_path:
'/opt/netapp/lib/ontap_mediator/ontap_mediator/server_config/intermediate.
srl'
```

- `cert_path` and `key_path` are server certificate variables.
- `ca_cert_path`, `ca_key_path`, and `ca_serial_path` are CA certificate variables.

Steps

1. Replace all `intermediate.*` files with the third-party certificates.
2. Create a certificate chain from the `intermediate.crt` and `ontap_mediator_server.crt` certificates:

   ```
   cat ontap_mediator_server.crt intermediate.crt >
   ontap_mediator_server_chain.crt
   ```

3. Update the `/opt/netapp/lib/ontap_mediator/uwsgi/ontap_mediator.ini` file.

   Update the values of `mediator_cert`, `mediator_key`, and `ca_certificate`:

   ```
   set-placeholder = mediator_cert =
   /opt/netapp/lib/ontap_mediator/ontap_mediator/server_config/ontap_mediator_ser
   ver_chain.crt

   set-placeholder = mediator_key =
   /opt/netapp/lib/ontap_mediator/ontap_mediator/server_config/ontap_mediator_ser
   ver.key

   set-placeholder = ca_certificate =
   /opt/netapp/lib/ontap_mediator/ontap_mediator/server_config/root_ca.crt
   ```

   - The `mediator_cert` value is the path of the `ontap_mediator_server_chain.crt` file.
   - The `mediator_key value` is the key path in the `ontap_mediator_server.crt` file, which is `ontap_mediator_server.key`.
   - The `ca_certificate` value is the path of the `root_ca.crt` file.
4. Verify that the following attributes of the newly generated certificates are set correctly:
   - Linux Group Owner: `netapp:netapp`
   - Linux permissions: `600`
5. Restart ONTAP Mediator:

   ```
   systemctl restart ontap_mediator
   ```

The certificate configuration is supplied to ONTAP Mediator in the configuration file located at `/opt/netapp/lib/ontap_mediator/ontap_mediator/server_config/ontap_mediator.config.yaml`. The file includes the following attributes:

```
cert_path:
'/opt/netapp/lib/ontap_mediator/ontap_mediator/server_config/ontap_mediato
r_server.crt'
key_path:
'/opt/netapp/lib/ontap_mediator/ontap_mediator/server_config/ontap_mediato
```

```
r_server.key'
ca_cert_path:
'/opt/netapp/lib/ontap_mediator/ontap_mediator/server_config/ca.crt'
ca_key_path:
'/opt/netapp/lib/ontap_mediator/ontap_mediator/server_config/ca.key'
ca_serial_path:
'/opt/netapp/lib/ontap_mediator/ontap_mediator/server_config/ca.srl'
```

- `cert_path` and `key_path` are server certificate variables.
- `ca_cert_path`, `ca_key_path`, and `ca_serial_path` are CA certificate variables.

Steps

1. Replace all `ca.*` files with the third-party certificates.
2. Create a certificate chain from the `ca.crt` and `ontap_mediator_server.crt` certificates:

   ```
   cat ontap_mediator_server.crt ca.crt > ontap_mediator_server_chain.crt
   ```

3. Update the `/opt/netapp/lib/ontap_mediator/uwsgi/ontap_mediator.ini` file.

   Update the values of `mediator_cert`, `mediator_key`, and `ca_certificate`:

   ```
   set-placeholder = mediator_cert =
   /opt/netapp/lib/ontap_mediator/ontap_mediator/server_config/ontap_mediator_ser
   ver_chain.crt

   set-placeholder = mediator_key =
   /opt/netapp/lib/ontap_mediator/ontap_mediator/server_config/ontap_mediator_ser
   ver.key

   set-placeholder = ca_certificate =
   /opt/netapp/lib/ontap_mediator/ontap_mediator/server_config/root_ca.crt
   ```

   - The `mediator_cert` value is the path of the `ontap_mediator_server_chain.crt` file.
   - The `mediator_key value` is the key path in the `ontap_mediator_server.crt` file, which is `ontap_mediator_server.key`.
   - The `ca_certificate` value is the path of the `root_ca.crt` file.
4. Verify that the following attributes of the newly generated certificates are set correctly:
   - Linux Group Owner: `netapp:netapp`
   - Linux permissions: `600`
5. Restart ONTAP Mediator:

   ```
   systemctl restart ontap_mediator
   ```

== Step 4: Optionally, use a different path or name for your third-party certificates

You can use third-party certificates with a different name other than `intermediate.*` or store the third-party certificates in a different location.

Steps

1. Configure the `/opt/netapp/lib/ontap_mediator/ontap_mediator/server_config/ontap_mediator.user_config.yaml` file to override the default variable values in the `ontap_mediator.config.yaml` file.

   If you obtained `intermediate.crt` from a PKI authority and you store its private key `intermediate.key` at the location `/opt/netapp/lib/ontap_mediator/ontap_mediator/server_config`, the `ontap_mediator.user_config.yaml` file should look like the following example:

   > If you used `intermediate.crt` to sign the `ontap_mediator_server.crt` certificate, the `intermediate.srl` file is generated. See Step 2: Generate a server certificate by signing with a third-party CA certification for more information.

```
[root@mediator_host server_config]# cat  ontap_mediator.user_config.yaml

# This config file can be used to override the default settings in
ontap_mediator.config.yaml
# To override a setting, copy the property key from
ontap_mediator.config.yaml to this file and
# set the property to the desired value. e.g.,
#
# The default value for 'default_mailboxes_per_target' is 4 in
ontap_mediator.config.yaml
#
# To override this value with 6 mailboxes per target, add the following
key/value pair
# below this comment:
#
# 'default_mailboxes_per_target': 6
#
cert_path:
'/opt/netapp/lib/ontap_mediator/ontap_mediator/server_config/ontap_media
tor_server.crt'
key_path:
'/opt/netapp/lib/ontap_mediator/ontap_mediator/server_config/ontap_media
tor_server.key'
ca_cert_path:
'/opt/netapp/lib/ontap_mediator/ontap_mediator/server_config/intermediat
e.crt'
ca_key_path:
'/opt/netapp/lib/ontap_mediator/ontap_mediator/server_config/intermediat
e.key'
ca_serial_path:
```

```
'/opt/netapp/lib/ontap_mediator/ontap_mediator/server_config/intermediat
e.srl'
```

a. If you are using a certificate structure where the `root_ca.crt` certificate provides an `intermediate.crt` certificate that signs the `ontap_mediator_server.crt` certificate, create a certificate chain from the `intermediate.crt` and `ontap_mediator_server.crt` certificates:

> You should have obtained the `intermediate.crt` and `ontap_mediator_server.crt` certificates from a PKI authority earlier in the procedure.

```
cat ontap_mediator_server.crt intermediate.crt >
ontap_mediator_server_chain.crt
```

b. Update the `/opt/netapp/lib/ontap_mediator/uvicorn/config.json` file.

Update the values of `ssl_keyfile`, `ssl_certfile`, and `ssl_ca_certs`:

```
ssl_keyfile:
/opt/netapp/lib/ontap_mediator/ontap_mediator/server_config/ontap_mediator_s
erver.key

ssl_certfile:
/opt/netapp/lib/ontap_mediator/ontap_mediator/server_config/ontap_mediator_s
erver_chain.crt

ssl_ca_certs:
/opt/netapp/lib/ontap_mediator/ontap_mediator/server_config/root_ca.crt
```

- The `ssl_keyfile` value is the key path in the `ontap_mediator_server.crt` file, which is `ontap_mediator_server.key`.
- The `ssl_certfile` value is the path of the `ontap_mediator_server_chain.crt` file.
- The `ssl_ca_certs` value is the path of the `root_ca.crt` file.

> For SnapMirror active sync clusters and MetroCluster IP configurations using ONTAP Mediator configured with HTTPS, you must add the `intermediate.crt` and `root_ca.crt` certificates to an ONTAP cluster. See Configure ONTAP Mediator and clusters for SnapMirror active sync and Configure the ONTAP Mediator for a MetroCluster IP configuration.

c. Verify that the following attributes of the newly generated certificates are set correctly:

- Linux Group Owner: `netapp:netapp`
- Linux permissions: `600`

2. Restart ONTAP Mediator when the certificates are updated in the configuration file:

```
systemctl restart ontap_mediator
```

You can use third-party certificates with a different name other than `intermediate.*` or store the third-party

certificates in a different location.

Steps

1. Configure the `/opt/netapp/lib/ontap_mediator/ontap_mediator/server_config/ontap_mediator.user_config.yaml` file to override the default variable values in the `ontap_mediator.config.yaml` file.

   If you obtained `intermediate.crt` from a PKI authority and you store its private key `intermediate.key` at the location `/opt/netapp/lib/ontap_mediator/ontap_mediator/server_config`, the `ontap_mediator.user_config.yaml` file should look like the following example:

   > If you used `intermediate.crt` to sign the `ontap_mediator_server.crt` certificate, the `intermediate.srl` file is generated. See Step 2: Generate a server certificate by signing with a third-party CA certification for more information.

```
[root@mediator_host server_config]# cat  ontap_mediator.user_config.yaml

# This config file can be used to override the default settings in
ontap_mediator.config.yaml
# To override a setting, copy the property key from
ontap_mediator.config.yaml to this file and
# set the property to the desired value. e.g.,
#
# The default value for 'default_mailboxes_per_target' is 4 in
ontap_mediator.config.yaml
#
# To override this value with 6 mailboxes per target, add the following
key/value pair
# below this comment:
#
# 'default_mailboxes_per_target': 6
#
cert_path:
'/opt/netapp/lib/ontap_mediator/ontap_mediator/server_config/ontap_media
tor_server.crt'
key_path:
'/opt/netapp/lib/ontap_mediator/ontap_mediator/server_config/ontap_media
tor_server.key'
ca_cert_path:
'/opt/netapp/lib/ontap_mediator/ontap_mediator/server_config/intermediat
e.crt'
ca_key_path:
'/opt/netapp/lib/ontap_mediator/ontap_mediator/server_config/intermediat
e.key'
ca_serial_path:
'/opt/netapp/lib/ontap_mediator/ontap_mediator/server_config/intermediat
```

```
e.srl'
```

a. If you are using a certificate structure where the `root_ca.crt` certificate provides an `intermediate.crt` certificate that signs the `ontap_mediator_server.crt` certificate, create a certificate chain from the `intermediate.crt` and `ontap_mediator_server.crt` certificates:

> You should have obtained the `intermediate.crt` and `ontap_mediator_server.crt` certificates from a PKI authority earlier in the procedure.

```
cat ontap_mediator_server.crt intermediate.crt >
ontap_mediator_server_chain.crt
```

b. Update the `/opt/netapp/lib/ontap_mediator/uwsgi/ontap_mediator.ini` file.

Update the values of `mediator_cert`, `mediator_key`, and `ca_certificate`:

```
set-placeholder = mediator_cert =
/opt/netapp/lib/ontap_mediator/ontap_mediator/server_config/ontap_mediator_s
erver_chain.crt

set-placeholder = mediator_key =
/opt/netapp/lib/ontap_mediator/ontap_mediator/server_config/ontap_mediator_s
erver.key

set-placeholder = ca_certificate =
/opt/netapp/lib/ontap_mediator/ontap_mediator/server_config/root_ca.crt
```

- The `mediator_cert` value is the path of the `ontap_mediator_server_chain.crt` file.
- The `mediator_key` value is the key path in the `ontap_mediator_server.crt` file, which is `ontap_mediator_server.key`.
- The `ca_certificate` value is the path of the `root_ca.crt` file.

> For SnapMirror Business Continuity (SM-BC) clusters, you must add the `intermediate.crt` and `root_ca.crt` certificates to an ONTAP cluster. See Configure ONTAP Mediator and clusters for SnapMirror active sync.

c. Verify that the following attributes of the newly generated certificates are set correctly:

- Linux Group Owner: `netapp:netapp`
- Linux permissions: `600`

2. Restart ONTAP Mediator when the certificates are updated in the configuration file:

```
systemctl restart ontap_mediator
```

You can use third-party certificates with a different name other than `ca.*` or store the third-party certificates in a different location.

Steps

1. Configure the `/opt/netapp/lib/ontap_mediator/ontap_mediator/server_config/ontap_mediator.user_config.yaml` file to override the default variable values in the `ontap_mediator.config.yaml` file.

   If you obtained `ca.crt` from a PKI authority and you store its private key `ca.key` at the location `/opt/netapp/lib/ontap_mediator/ontap_mediator/server_config`, the `ontap_mediator.user_config.yaml` file should look like the following example:

   > If you used `ca.crt` to sign the `ontap_mediator_server.crt` certificate, the `ca.srl` file is generated. See Step 2: Generate a server certificate by signing with a third-party CA certification for more information.

```
[root@mediator_host server_config]# cat  ontap_mediator.user_config.yaml

# This config file can be used to override the default settings in
ontap_mediator.config.yaml
# To override a setting, copy the property key from
ontap_mediator.config.yaml to this file and
# set the property to the desired value. e.g.,
#
# The default value for 'default_mailboxes_per_target' is 4 in
ontap_mediator.config.yaml
#
# To override this value with 6 mailboxes per target, add the following
key/value pair
# below this comment:
#
# 'default_mailboxes_per_target': 6
#
cert_path:
'/opt/netapp/lib/ontap_mediator/ontap_mediator/server_config/ontap_media
tor_server.crt'
key_path:
'/opt/netapp/lib/ontap_mediator/ontap_mediator/server_config/ontap_media
tor_server.key'
ca_cert_path:
'/opt/netapp/lib/ontap_mediator/ontap_mediator/server_config/ca.crt'
ca_key_path:
'/opt/netapp/lib/ontap_mediator/ontap_mediator/server_config/ca.key'
ca_serial_path:
'/opt/netapp/lib/ontap_mediator/ontap_mediator/server_config/ca.srl'
```

   a. If you are using a certificate structure where the `root_ca.crt` certificate provides an `ca.crt` certificate that signs the `ontap_mediator_server.crt` certificate, create a certificate chain from the `ca.crt` and `ontap_mediator_server.crt` certificates:

> You should have obtained the `ca.crt` and `ontap_mediator_server.crt` certificates from a PKI authority earlier in the procedure.

```
cat ontap_mediator_server.crt ca.crt > ontap_mediator_server_chain.crt
```

b. Update the `/opt/netapp/lib/ontap_mediator/uwsgi/ontap_mediator.ini` file.

Update the values of `mediator_cert`, `mediator_key`, and `ca_certificate`:

```
set-placeholder = mediator_cert =
/opt/netapp/lib/ontap_mediator/ontap_mediator/server_config/ontap_mediator_s
erver_chain.crt

set-placeholder = mediator_key =
/opt/netapp/lib/ontap_mediator/ontap_mediator/server_config/ontap_mediator_s
erver.key

set-placeholder = ca_certificate =
/opt/netapp/lib/ontap_mediator/ontap_mediator/server_config/root_ca.crt
```

- The `mediator_cert` value is the path of the `ontap_mediator_server_chain.crt` file.
- The `mediator_key` value is the key path in the `ontap_mediator_server.crt` file, which is `ontap_mediator_server.key`.
- The `ca_certificate` value is the path of the `root_ca.crt` file.

> For SnapMirror Business Continuity (SM-BC) clusters, you must add the `ca.crt` and `root_ca.crt` certificates to an ONTAP cluster. See Configure ONTAP Mediator and clusters for SnapMirror active sync.

c. Verify that the following attributes of the newly generated certificates are set correctly:

- Linux Group Owner: `netapp:netapp`
- Linux permissions: `600`

2. Restart ONTAP Mediator when the certificates are updated in the configuration file:

```
systemctl restart ontap_mediator
```

= Switch ONTAP Mediator to the legacy CA certificate common name
:icons: font
:relative_path: ./administer/
:imagesdir: /tmp/d20261109-3041152-j9lm3i/source/./administer/../media/

Beginning with ONTAP Mediator 1.13, the self-signed CA certificates include a per-installation identifier in their common name. If your ONTAP Mediator instance must pair with an ONTAP release that does not recognize this identifier, switch to the legacy fixed common name.

By default, the ONTAP Mediator CA certificates carry a per-installation identifier in their common name,

for example `ONTAP Mediator CA 1a2b3c4d`. This enables you to register several ONTAP Mediator instances with the same ONTAP cluster, because ONTAP refuses to install two CA certificates that share a subject name but hold different keys.

The following ONTAP releases locate the ONTAP Mediator CA by matching the common name exactly and do not recognize the identifier:

- ONTAP 9.11.1 and earlier
- ONTAP patch releases earlier than 9.12.1P15, 9.13.1P12, 9.14.1P8, and 9.15.1P2

If an ONTAP Mediator instance must pair with one of these releases, use the legacy fixed common name.

About this task

- Perform this task on the Linux host where you installed ONTAP Mediator.
- While the legacy common name is in effect, you can register only one ONTAP Mediator instance per ONTAP cluster.

Steps

1. Add the following line to the `/opt/netapp/lib/ontap_mediator/ontap_mediator/server_config/ontap_mediator.user_config.yaml` file. This file is preserved across ONTAP Mediator upgrades.

   ```
   'generate_unique_ca_name': false
   ```

2. Regenerate the certificates:

   ```
   [root@localhost ~]# cd
   /opt/netapp/lib/ontap_mediator/ontap_mediator/server_config/;
   ./make_self_signed_certs.sh overwrite
   ```

3. Restart the service:

   ```
   [root@localhost ~]# sudo systemctl restart ontap_mediator
   ```

4. On every paired ONTAP cluster where self-signed CA certificates are used, perform the following actions:
   a. Remove ONTAP Mediator.
   b. Delete the previously installed CA certificates.
   c. Install the regenerated `root_ca.crt` and `intermediate.crt` certificates.
   d. Add ONTAP Mediator again.

   Regeneration always produces new keys, so you must delete the old CA certificates first. If you leave them in place, ONTAP retains a CA that no longer signs the server certificate. Newer ONTAP releases also reject the installation of the replacement CA certificates because the new and old certificates share the same subject name.

= Troubleshoot ONTAP Mediator certificate-related issues
:icons: font
:relative_path: ./administer/
:imagesdir: /tmp/d20261109-3041152-j9lm3i/source/./administer/../media/

Use these checks to verify certificate properties when troubleshooting certificate issues.

== Verify certificate expiration

Use the following command to identify the certificate validity range.

```
[root@mediator_host server_config]# openssl x509 -in intermediate.crt
-text -noout
Certificate:
    Data:
...
        Validity
            Not Before: Feb 22 19:57:25 2024 GMT
            Not After : Feb 15 19:57:25 2029 GMT
```

```
[root@mediator_host server_config]# openssl x509 -in ca.crt -text -noout
Certificate:
    Data:
...
        Validity
            Not Before: Feb 22 19:57:25 2024 GMT
            Not After : Feb 15 19:57:25 2029 GMT
```

== Verify X509v3 extensions in CA certification

Use the following command to verify the X509v3 extensions in the CA certification.

The properties defined within **`v3_ca`** in `openssl_ca.cnf` are displayed as `X509v3 extensions` in `intermediate.crt`.

```
[root@mediator_host server_config]# pwd
/opt/netapp/lib/ontap_mediator/ontap_mediator/server_config

[root@mediator_host server_config]# cat openssl_ca.cnf
...
[ v3_ca ]
subjectKeyIdentifier = hash
```

```
authorityKeyIdentifier = keyid:always,issuer
basicConstraints = critical, CA:true
keyUsage = critical, cRLSign, digitalSignature, keyCertSign

[root@mediator_host server_config]# openssl x509 -in intermediate.crt
-text -noout
Certificate:
    Data:
...
        X509v3 extensions:
            X509v3 Subject Key Identifier:

9F:06:FA:47:00:67:BA:B2:D4:82:70:38:B8:48:55:B5:24:DB:FC:27
            X509v3 Authority Key Identifier:

keyid:9F:06:FA:47:00:67:BA:B2:D4:82:70:38:B8:48:55:B5:24:DB:FC:27

            X509v3 Basic Constraints: critical
                CA:TRUE
            X509v3 Key Usage: critical
                Digital Signature, Certificate Sign, CRL Sign
```

The properties defined within **v3_ca** in `openssl_ca.cnf` are displayed as `X509v3 extensions` in `ca.crt`.

```
[root@mediator_host server_config]# pwd
/opt/netapp/lib/ontap_mediator/ontap_mediator/server_config

[root@mediator_host server_config]# cat openssl_ca.cnf
...
[ v3_ca ]
subjectKeyIdentifier = hash
authorityKeyIdentifier = keyid:always,issuer
basicConstraints = critical, CA:true
keyUsage = critical, cRLSign, digitalSignature, keyCertSign

[root@mediator_host server_config]# openssl x509 -in ca.crt -text -noout
Certificate:
    Data:
...
        X509v3 extensions:
            X509v3 Subject Key Identifier:

9F:06:FA:47:00:67:BA:B2:D4:82:70:38:B8:48:55:B5:24:DB:FC:27
            X509v3 Authority Key Identifier:
```

```
keyid:9F:06:FA:47:00:67:BA:B2:D4:82:70:38:B8:48:55:B5:24:DB:FC:27

            X509v3 Basic Constraints: critical
                CA:TRUE
            X509v3 Key Usage: critical
                Digital Signature, Certificate Sign, CRL Sign
```

== Verify X509v3 extensions in server certificate and subject Alt Names

The `v3_req` properties defined in the `openssl_server.cnf` configuration file are displayed as `X509v3 extensions` in the certificate.

In the following example, you can obtain the variables in the `alt_names` sections by running the commands `hostname -A` and `hostname -I` on the Linux VM on which ONTAP Mediator is installed.

Check with your network administrator for the correct values of the variables.

```
[root@mediator_host server_config]# pwd
/opt/netapp/lib/ontap_mediator/ontap_mediator/server_config

[root@mediator_host server_config]# cat openssl_server.cnf
...
[ v3_req ]
basicConstraints       = CA:false
extendedKeyUsage       = serverAuth
keyUsage               = keyEncipherment, dataEncipherment
subjectAltName         = @alt_names

[ alt_names ]
DNS.1 = abc.company.com
DNS.2 = abc-v6.company.com
IP.1 = 1.2.3.4
IP.2 = abcd:abcd:abcd:abcd:abcd:abcd

[root@mediator_host server_config]# openssl x509 -in intermediate.crt
-text -noout
Certificate:
    Data:
...

        X509v3 extensions:
            X509v3 Basic Constraints:
                CA:FALSE
            X509v3 Extended Key Usage:
                TLS Web Server Authentication
```

```
            X509v3 Key Usage:
                Key Encipherment, Data Encipherment
            X509v3 Subject Alternative Name:
                DNS:abc.company.com, DNS:abc-v6.company.com, IP
Address:1.2.3.4, IP Address:abcd:abcd:abcd:abcd:abcd:abcd
```

```
[root@mediator_host server_config]# pwd
/opt/netapp/lib/ontap_mediator/ontap_mediator/server_config

[root@mediator_host server_config]# cat openssl_server.cnf
...
[ v3_req ]
basicConstraints       = CA:false
extendedKeyUsage       = serverAuth
keyUsage               = keyEncipherment, dataEncipherment
subjectAltName         = @alt_names

[ alt_names ]
DNS.1 = abc.company.com
DNS.2 = abc-v6.company.com
IP.1 = 1.2.3.4
IP.2 = abcd:abcd:abcd:abcd:abcd:abcd

[root@mediator_host server_config]# openssl x509 -in ca.crt -text -noout
Certificate:
    Data:
...

        X509v3 extensions:
            X509v3 Basic Constraints:
                CA:FALSE
            X509v3 Extended Key Usage:
                TLS Web Server Authentication
            X509v3 Key Usage:
                Key Encipherment, Data Encipherment
            X509v3 Subject Alternative Name:
                DNS:abc.company.com, DNS:abc-v6.company.com, IP
Address:1.2.3.4, IP Address:abcd:abcd:abcd:abcd:abcd:abcd
```

== Verify that a private key matches with a certificate

You can verify whether a particular private key matches with a certificate.

Use the following OpenSSL commands on the key and certificate respectively.

```
[root@mediator_host server_config]# openssl rsa -noout -modulus -in
intermediate.key | openssl md5
Enter pass phrase for intermediate.key:
(stdin)= 14c6b98b0c7c59012b1de89eee4a9dbc
[root@mediator_host server_config]# openssl x509 -noout -modulus -in
intermediate.crt | openssl md5
(stdin)= 14c6b98b0c7c59012b1de89eee4a9dbc
```

```
[root@mediator_host server_config]# openssl rsa -noout -modulus -in ca.key
| openssl md5
Enter pass phrase for ca.key:
(stdin)= 14c6b98b0c7c59012b1de89eee4a9dbc
[root@mediator_host server_config]# openssl x509 -noout -modulus -in
ca.crt | openssl md5
(stdin)= 14c6b98b0c7c59012b1de89eee4a9dbc
```

If the `-modulus` attribute for both match, it indicates that the private key and certificate pair are compatible and can work with each other.

== Verify that a server certificate is created from a particular CA certificate

You can use the following command to verify that the server certificate is created from a particular CA certificate.

```
[root@mediator_host server_config]# openssl verify -CAfile root_ca.crt
--untrusted intermediate.crt ontap_mediator_server.crt
ontap_mediator_server.crt: OK
[root@mediator_host server_config]#
```

```
[root@mediator_host server_config]# openssl verify -CAfile ca.crt
ontap_mediator_server.crt
ontap_mediator_server.crt: OK
```

If the Online Certificate Status Protocol (OCSP) validation is being used, use the command openssl-verify.

= Maintain the host OS for ONTAP Mediator
:icons: font
:relative_path: ./administer/
:imagesdir: /tmp/d20261109-3041152-j9lm3i/source/./administer/../media/

Ensure the host OS for ONTAP Mediator is maintained regularly to achieve optimal performance.

== Reboot the host

Reboot the host only when clusters are healthy. Clusters cannot respond to failures while ONTAP Mediator is offline. Set a time for maintenance before you reboot.

ONTAP Mediator automatically resumes during a reboot and re-enters previously configured relationships with ONTAP clusters.

== Host package updates

Update any library or yum package except the kernel. Reboot the host if required for changes to take effect. Schedule a service window before rebooting the host.

If you install the `yum-utils` package, use the `needs-restarting` command to detect if any package changes require a reboot.

Reboot after updating ONTAP Mediator dependencies because changes do not take immediate effect.

== Upgrade the host OS kernel

SCST must be compiled for the kernel you are using. To update the OS, you need to schedule time for maintenance.

Before you begin

The following requirements apply when the SCST package is installed with ONTAP Mediator.

- You can update all library packages except the kernel, but you might need to reboot to apply changes in ONTAP Mediator. Schedule downtime if you need to reboot.
- You should keep the OS kernel up to date. Upgrade the kernel core to a supported version listed in the ONTAP Mediator version matrix. You must reboot the system, so plan a maintenance window for the outage.
  - Uninstall the SCST kernel module before you reboot, and then reinstall it afterwards.
  - Prepare a supported version of SCST to reinstall before you start the kernel OS upgrade.

> If you installed ONTAP Mediator 1.12 or later with support for HTTPS only, the SCST package is not installed and the host OS kernel can be upgraded without affecting ONTAP Mediator. Refer to your host OS documentation for upgrade instructions.

Steps

Follow these steps to upgrade the host OS kernel.

> Before upgrading the kernel, check that the OS and ONTAP Mediator version are compatible. For supported versions, see the OS support matrix.

1. Stop ONTAP Mediator.
2. Uninstall the SCST package, see [Perform host maintenance]. (SCST doesn't provide an upgrade mechanism.)
3. Upgrade the OS, and reboot.

4. Re-install the SCST package.
5. Re-enable ONTAP Mediator.

== Perform host maintenance

Upgrading the VM kernel can cause compatibility issues with SCST modules. Manually uninstall and reinstall SCST.

If you installed ONTAP Mediator 1.12 or later with support for HTTPS only, the SCST package is not installed and you can skip this task.

=== Step 1: Uninstall SCST

To uninstall SCST, use the tar bundle for your ONTAP Mediator version.

Steps

1. Download the appropriate SCST bundle (as shown in the following table) and extract it.

For this version …	Use this tar bundle…
ONTAP Mediator 1.13	scst-3.10.tar.gz
ONTAP Mediator 1.12.1	scst-3.10.tar.gz
ONTAP Mediator 1.12	scst-3.10.tar.gz
ONTAP Mediator 1.11	scst-3.9.tar.gz
ONTAP Mediator 1.10	scst-3.9.tar.gz
ONTAP Mediator 1.9.1	scst-3.8.0.tar.bz2
ONTAP Mediator 1.9	scst-3.8.0.tar.bz2
ONTAP Mediator 1.8	scst-3.8.0.tar.bz2
ONTAP Mediator 1.7	scst-3.7.0.tar.bz2
ONTAP Mediator 1.6	scst-3.7.0.tar.bz2
ONTAP Mediator 1.5	scst-3.6.0.tar.bz2
ONTAP Mediator 1.4	scst-3.6.0.tar.bz2
ONTAP Mediator 1.3	scst-3.5.0.tar.bz2
ONTAP Mediator 1.1	scst-3.4.0.tar.bz2
ONTAP Mediator 1.0	scst-3.3.0.tar.bz2

   a. Access the open source package from the SCST sourceforge downloads.
   b. Select **Download released versions**.
   c. Extract the bundle to your VM.

2. Run the following uninstall commands in the `scst` directory:

   a. `systemctl stop mediator-scst`
   b. `make scstadm_uninstall`
   c. `make iscsi_uninstall`

d. `make usr_uninstall`

e. `make scst_uninstall`

f. `depmod`

=== Step 2: Install SCST

To manually install SCST, you need the SCST tar bundle that is used for the installed version of ONTAP Mediator (see the SCST table).

About this task

- Perform this step before you install the ONTAP Mediator. If the SCST version you're using is newer than the version bundled with the ONTAP Mediator installer, the installer skips this step.
- In some scenarios, the default installation of SCST on the upgraded OS can fail because the upgraded OS requires a patch.

  Beginning with ONTAP Mediator 1.12.1, if the installation fails, you can use the SCST repair tool located in `/usr/local/bin/mediator_repair_scst` to repair the installation.

  If the upgraded OS requires a patch, the tool contains it for the version of SCST shipped within the ONTAP Mediator package. The `mediator_repair_scst` tool is a standalone tool that is separate from the main installer and is packaged inside the ONTAP Mediator `tgz` file. It does not affect the ONTAP Mediator installation or upgrade paths.

Steps

1. Run the following install commands in the `scst` directory:

   a. `make 2release`

   b. `make scst_install`

   c. `make usr_install`

   d. `make iscsi_install`

   e. `make scstadm_install`

   f. `depmod`

If you're performing a first-time installation and want to pre-install ONTAP Mediator, run the following command before continuing with the next step:

```
mkdir -p /opt/netapp/lib/ontap_mediator/ontap_mediator/SCST_mod_keys
```

a. `cp scst/src/certs/scst_module_key.der /opt/netapp/lib/ontap_mediator/ontap_mediator/SCST_mod_keys/`

b. `patch /etc/init.d/scst < /opt/netapp/lib/ontap_mediator/systemd/scst.patch`

> If you pre-install SCST before ONTAP Mediator during a first-time installation, skip this step. The installer applies relevant SCST patches.

1. Optionally, if Secure Boot is enabled, before you reboot, perform the following steps:

c. Determine each file name for the `scst_vdisk`, `scst`, and `iscsi_scst` modules:

```
[root@localhost ~]# modinfo -n scst_vdisk
[root@localhost ~]# modinfo -n scst
[root@localhost ~]# modinfo -n iscsi_scst
```

d. Determine the kernel release:

```
[root@localhost ~]# uname -r
```

e. Sign each module file with the kernel:

```
[root@localhost ~]# /usr/src/kernels/<KERNEL-RELEASE>/scripts/sign-
file \sha256 \
/opt/netapp/lib/ontap_mediator/ontap_mediator/SCST_mod_keys/scst_mod
ule_key.priv \
/opt/netapp/lib/ontap_mediator/ontap_mediator/SCST_mod_keys/scst_mod
ule_key.der \
_module-filename_
```

f. Install the UEFI key with the firmware.

Instructions for installing the UEFI key are located at:

`/opt/netapp/lib/ontap_mediator/ontap_mediator/SCST_mod_keys/README.module-signing`

The generated UEFI key is located at:

`/opt/netapp/lib/ontap_mediator/ontap_mediator/SCST_mod_keys/scst_module_key.der`

1. Reboot the system:

```
reboot
```

== Host changes to the hostname or IP

About this task

- Perform this task on the Linux host where you installed ONTAP Mediator.
- Do this only if the self-signed certificates are no longer valid because the hostname or IP address changes after you install ONTAP Mediator.
- After the temporary self-signed certificate has been replaced by a trusted third-party certificate, you do *not* use this task to regenerate a certificate. If you do not have a self-signed certificate, you cannot use this procedure.

Step

Create a temporary self-signed certificate for the current host:

1. Restart ONTAP Mediator:

   ```
   ./make_self_signed_certs.sh overwrite
   ```

```
[root@xyz000123456 ~]# cd
/opt/netapp/lib/ontap_mediator/ontap_mediator/server_config
[root@xyz000123456 server_config]# ./make_self_signed_certs.sh
overwrite

Adding Subject Alternative Names to the self-signed server
certificate
#
# OpenSSL example configuration file.
Generating self-signed certificates
Generating RSA private key, 4096 bit long modulus (2 primes)
.....................................................................
.....................................................................
.........................++++
.....................................................++++
e is 65537 (0x010001)
Generating a RSA private key
................................................++++
.....................................................................
.....................................................................
.....++++
writing new private key to 'ontap_mediator_server.key'
-----
Signature ok
subject=C = US, ST = California, L = San Jose, O = "NetApp, Inc.",
OU = ONTAP Core Software, CN = ONTAP Mediator, emailAddress =
support@netapp.com
Getting CA Private Key

[root@xyz000123456 server_config]# systemctl restart ontap_mediator
```

= Uninstall ONTAP Mediator
:icons: font
:relative_path: ./administer/
:imagesdir: /tmp/d20261109-3041152-j9lm3i/source/./administer/../media/

If necessary, uninstall ONTAP Mediator after disconnecting it from ONTAP.

Before you begin

You must disconnect ONTAP Mediator from ONTAP before removing it.

About this task

Perform this task on the Linux host where you installed ONTAP Mediator.

If you are unable to reach this command, you might need to run the command using the full path as shown in the following example:

```
/usr/local/bin/uninstall_ontap_mediator
```

Step

1. Uninstall ONTAP Mediator:

```
uninstall_ontap_mediator
```

```
 [root@mediator-host ~]# uninstall_ontap_mediator

 ONTAP Mediator: Self Extracting Uninstaller

 + Removing ONTAP Mediator. (Log:
/tmp/ontap_mediator.GmRGdA/uninstall_ontap_mediator/remove.log)
 + Remove successful.
 [root@mediator-host ~]#
```

= Legal notices
:icons: font
:relative_path: ./
:imagesdir: /tmp/d20261109-3041152-j9lm3i/source/./administer/./media/

Legal notices provide access to copyright statements, trademarks, patents, and more.

== Copyright

https://www.netapp.com/company/legal/copyright/

== Trademarks

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.

https://www.netapp.com/company/legal/trademarks/

== Patents

A current list of NetApp owned patents can be found at:

https://www.netapp.com/pdf.html?item=/media/11887-patentspage.pdf

== Privacy policy

https://www.netapp.com/company/legal/privacy-policy/

== Open source

Notice files provide information about third-party copyright and licenses used in NetApp software.

== ONTAP Mediator for MetroCluster IP and SnapMirror active sync

- Notice for ONTAP Mediator 1.13
- Notice for ONTAP Mediator 1.12.1
- Notice for ONTAP Mediator 1.11
- Notice for ONTAP Mediator 1.10
- Notice for ONTAP Mediator 1.9.1
- Notice for ONTAP Mediator 1.8
- Notice for ONTAP Mediator 1.7
- Notice for ONTAP Mediator 1.5

Copyright information

Copyright © 2026 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.

LIMITED RIGHTS LEGEND: Use, duplication, or disclosure by the government is subject to restrictions as set forth in subparagraph (b)(3) of the Rights in Technical Data -Noncommercial Items at DFARS 252.227-7013 (FEB 2014) and FAR 52.227-19 (DEC 2007).

Data contained herein pertains to a commercial product and/or commercial service (as defined in FAR 2.101) and is proprietary to NetApp, Inc. All NetApp technical data and computer software provided under this Agreement is commercial in nature and developed solely at private expense. The U.S. Government has a non-exclusive, non-transferrable, nonsublicensable, worldwide, limited irrevocable license to use the Data only in connection with and in support of the U.S. Government contract under which the Data was delivered. Except as provided herein, the Data may not be used, disclosed, reproduced, modified, performed, or displayed without the prior written approval of NetApp, Inc. United States Government license rights for the Department of Defense are limited to those rights identified in DFARS clause 252.227-7015(b) (FEB 2014).

Trademark information

NETAPP, the NETAPP logo, and the marks listed at http://www.netapp.com/TM are trademarks of NetApp, Inc. Other company and product names may be trademarks of their respective owners.