



Administer ONTAP Mediator

ONTAP Mediator

NetApp
October 19, 2026

This PDF was generated from <https://docs.netapp.com/us-en/ontap-mediator/administer/upgrade-ontap-mediator-task.html> on October 19, 2026. Always check docs.netapp.com for the latest.

Table of Contents

- Administer ONTAP Mediator 1
 - Upgrade ONTAP Mediator 1
 - Upgrade ONTAP Mediator 1
 - Upgrade the host OS and reinstall or upgrade ONTAP Mediator 9
- Configure ONTAP Mediator attributes 14
 - ONTAP Mediator configurable settings 14
 - Install third-party SSL certificates 15
 - Configure password attack protection 17
 - Configure password complexity rules 17
 - Configure the free-space threshold 18
 - Configure reserved log space 18

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-1.12.1/ontap-mediator-1.12.1 -y` command to skip the installer prompts.

Steps

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

```
./ontap-mediator-1.12.1/ontap-mediator-1.12.1
```



To skip the signature check during the upgrade, use the `./ontap-mediator-1.12.1/ontap-mediator-1.12.1 --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-1.11.0/ontap-mediator-1.11.0

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
+ Unpacking the ONTAP Mediator installer
ONTAP Mediator requires two user accounts. One for the service
(netapp), and one for use by ONTAP to the mediator API
(mediatoradmin).
Using default account names: netapp + mediatoradmin
```

Enter ONTAP Mediator user account (mediatoradmin) password:

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

+ Checking if SELinux is in enforcing mode

+ Checking for default Linux firewall

+ Installing required packages.

Last metadata expiration check: 0:03:56 ago on Mon 20 Apr 2026
02:49:08 PM EDT.

Package openssl-1:3.5.1-7.el9_7.x86_64 is already installed.

Package openssl-devel-1:3.5.1-7.el9_7.x86_64 is already installed.

Package kernel-devel-5.14.0-611.45.1.el9_7.x86_64 is already
installed.

Package kernel-modules-core-5.14.0-611.45.1.el9_7.x86_64 is already
installed.

Package gcc-11.5.0-11.el9.x86_64 is already installed.

Package make-1:4.3-8.el9.x86_64 is already installed.

Package which-2.21-30.el9_6.x86_64 is already installed.

Package libselinux-utils-3.6-3.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 bzip2-1.0.8-10.el9_5.x86_64 is already installed.

Package efibootmgr-16-12.el9.x86_64 is already installed.

Package mokutil-2:0.7.2-1.el9.x86_64 is already installed.

Package python3-pip-21.3.1-1.el9.noarch is already installed.

Package elfutils-libelf-devel-0.193-1.el9.x86_64 is already installed.

Package policycoreutils-python-utils-3.6-3.el9.noarch is already
installed.

Package python3.12-3.12.12-4.el9_7.2.x86_64 is already installed.

Package python3.12-devel-3.12.12-4.el9_7.2.x86_64 is already
installed.

Dependencies resolved.

Nothing to do.

Complete!

OS package installations finished

+ Installing ONTAP Mediator. (Log: /root/ontap_mediator.uigddi/ontap-

```
mediator-1.11.0/ontap-mediator-1.11.0/install_20260420145003.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_20260420145003.log)

+ 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 ~]# ontap-mediator-1.12.1/ontap-mediator-1.12.1
```

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

+ 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 support for HTTPS 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 1.11.0 to 1.12.1.

```

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/scs001097036-1_1776711598_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: 0d5dcf349b27, Head revision: 758e4873c40d
Performing database migrations against clone...
Migrations successful.
Database upgrade operation complete.
+ Adding python312 specific packages
+ Upgrading ONTAP Mediator. (Log: /root/ontap_mediator.gwaGmU/ontap-
mediator-1.12.1/ontap-mediator-1.12.1/upgrade_20260420145942.log)

+ 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-1.11.0/ontap-mediator-1.11.0

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
+ Unpacking the ONTAP Mediator installer
ONTAP Mediator requires two user accounts. One for the service
(netapp), and one for use by ONTAP to the mediator API
(mediatoradmin).
Using default account names: netapp + mediatoradmin
```

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

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

```
+ Checking if SELinux is in enforcing mode
```

```
+ Checking for default Linux firewall
```

```
+ Installing required packages.
```

```
Last metadata expiration check: 0:33:41 ago on Tue 21 Apr 2026
10:48:35 AM EDT.
```

```
Package openssl-1:3.5.1-7.el9_7.x86_64 is already installed.
```

```
Package openssl-devel-1:3.5.1-7.el9_7.x86_64 is already installed.
```

```
Package kernel-devel-5.14.0-611.45.1.el9_7.x86_64 is already
installed.
```

```
Package kernel-modules-core-5.14.0-611.45.1.el9_7.x86_64 is already
installed.
```

```
Package gcc-11.5.0-11.el9.x86_64 is already installed.
```

```
Package make-1:4.3-8.el9.x86_64 is already installed.
```

```
Package which-2.21-30.el9_6.x86_64 is already installed.
```

```
Package libselinux-utils-3.6-3.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 bzip2-1.0.8-10.el9_5.x86_64 is already installed.
```

```
Package efibootmgr-16-12.el9.x86_64 is already installed.
```

```
Package mokutil-2:0.7.2-1.el9.x86_64 is already installed.
```

```
Package python3-pip-21.3.1-1.el9.noarch is already installed.
```

```

Package elfutils-libelf-devel-0.193-1.el9.x86_64 is already installed.
Package policycoreutils-python-utils-3.6-3.el9.noarch is already
installed.
Package python3.12-3.12.12-4.el9_7.2.x86_64 is already installed.
Package python3.12-devel-3.12.12-4.el9_7.2.x86_64 is already
installed.
Dependencies resolved.
Nothing to do.
Complete!
OS package installations finished
+ Installing ONTAP Mediator. (Log: /root/ontap_mediator.JclgnS/ontap-
mediator-1.11.0/ontap-mediator-1.11.0/install_20260421112205.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_20260421112205.log)

+ 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~]#
[root@mediator_host~]# tar -zxvf ontap-mediator-1.12.1.tgz
ontap-mediator-1.12.1/
ontap-mediator-1.12.1/csc-prod-chain-ONTAP-Mediator.pem
ontap-mediator-1.12.1/tsa-prod-ONTAP-Mediator.pem
ontap-mediator-1.12.1/csc-prod-ONTAP-Mediator.pem
ontap-mediator-1.12.1/tsa-prod-chain-ONTAP-Mediator.pem
ontap-mediator-1.12.1/ONTAP-Mediator-production.pub
ontap-mediator-1.12.1/ontap-mediator-1.12.1
ontap-mediator-1.12.1/ontap-mediator-1.12.1.sig
ontap-mediator-1.12.1/ontap-mediator-1.12.1.tsr
ontap-mediator-1.12.1/ontap-mediator-1.12.1.sig.tsr

```

```

[root@mediator_host~]#
[root@mediator_host~]# ontap-mediator-1.12.1/ontap-mediator-1.12.1
--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
+ 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
ONTAP Mediator will be upgraded from version 1.11.0 to 1.12.1.

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/scs001097036-1_1776785423_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: 0d5dcf349b27, Head revision: 758e4873c40d
Performing database migrations against clone...
Migrations successful.
Database upgrade operation complete.
+ Adding python312 specific packages
+ Upgrading ONTAP Mediator. (Log: /root/ontap_mediator.cRy0mN/ontap-
mediator-1.12.1/ontap-mediator-1.12.1/upgrade_20260421113003.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_20260421113003.log)\n

```

```
+ 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:

```
[root@mediator-host ~]# tar -czf ontap_mediator_file_backup.tgz -C
/opt/netapp/lib/ontap_mediator ./log
./ontap_mediator/server_config/ontap_mediator.user_config.yaml
[root@mediator-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
Job ID Name              Owing
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
....

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
....

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
```

Job	ID	Name	Owning		State
			Vserver	Node	
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 `low_space_threshold_mib: 10` 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
<code>cert_path</code>	<code>/opt/netapp/lib/ontap_mediator/ontap_mediator/server_config/ontap_mediator_server.crt</code>
<code>key_path</code>	<code>/opt/netapp/lib/ontap_mediator/ontap_mediator/server_config/ontap_mediator_server.key</code>
<code>ca_cert_path</code>	<code>/opt/netapp/lib/ontap_mediator/ontap_mediator/server_config/intermediate.crt</code>
<code>ca_key_path</code>	<code>/opt/netapp/lib/ontap_mediator/ontap_mediator/server_config/intermediate.key</code>
<code>ca_serial_path</code>	<code>/opt/netapp/lib/ontap_mediator/ontap_mediator/server_config/intermediate.srl</code>
<code>cert_valid_days</code>	1095
<code>x509_passin_pwd</code>	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

Variable	Path
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.



Toggleing 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/d20261019-6444-4nnphz/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_username
```

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/d20261019-6444-4nnphz/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:

```
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:

```
systemctl start mediator-scst
```

3. Start ONTAP Mediator:

```
systemctl start ontap_mediator
```

= Verify ONTAP Mediator health :icons: font :relative_path: ./administer/ :imagesdir: /tmp/d20261019-6444-4nnphz/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:

```
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/d20261019-6444-4nnphz/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/d20261019-6444-4nnphz/source/./administer/./media/

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

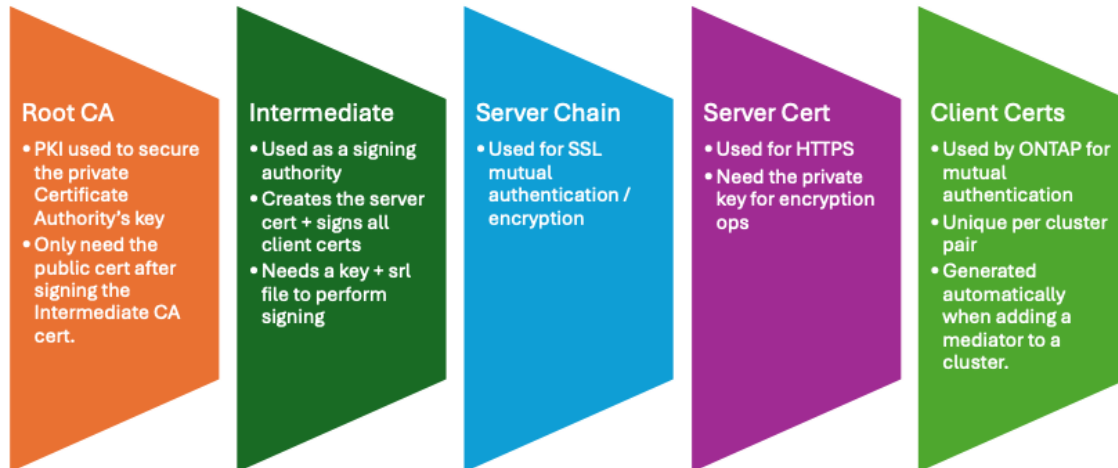


- Third-party certificates are only supported beginning with ONTAP 9.16.1 and in some earlier ONTAP patch releases. See [NetApp Bugs Online Bug ID CONTAP-243278](#).
- Third-party certificates are only supported on systems running ONTAP Mediator 1.7 or later.

About this task

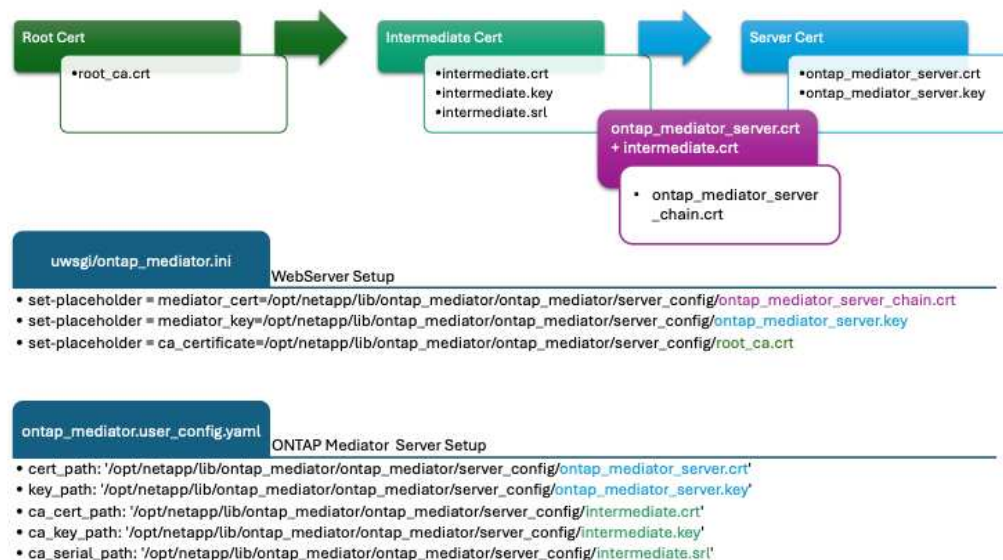
- Perform this task on the Linux host where you installed ONTAP Mediator.
- You can perform this task if the generated self-signed certificates need to be replaced by certificates obtained from a trusted subordinate certificate authority (CA). To accomplish this, you should have access to a trusted public-key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

ONTAP Mediator Certificate Purposes



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

ONTAP Mediator Certificates



== 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.
cert'
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/unicorn/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_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'
```

- `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_server_chain.crt
```

```
set-placeholder = mediator_key =
/opt/netapp/lib/ontap_mediator/ontap_mediator/server_config/ontap_mediator_server.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_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_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_server_chain.crt

set-placeholder = mediator_key =
```

```
/opt/netapp/lib/ontap_mediator/ontap_mediator/server_config/ontap_mediator_server.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_medi
tor_server.crt'
key_path:
'/opt/netapp/lib/ontap_mediator/ontap_mediator/server_config/ontap_medi
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/unicorn/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_medi
tor_server.crt'
key_path:
'/opt/netapp/lib/ontap_mediator/ontap_mediator/server_config/ontap_medi
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_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'
```

- 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_server_chain.crt
```

```
set-placeholder = mediator_key =
/opt/netapp/lib/ontap_mediator/ontap_mediator/server_config/ontap_mediator_server.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](#)

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
```

= Troubleshoot ONTAP Mediator certificate-related issues :icons: font :relative_path: ./administer/
:imagesdir: /tmp/d20261019-6444-4nnphz/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/d20261019-6444-4nnphz/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 1.12.1	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/d20261019-6444-4nnphz/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 ~]#
```

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.