Mediator service for MetroCluster and SnapMirror active sync

ONTAP 9

NetApp
June 04, 2024

Table of Contents

Mediator service for MetroCluster and SnapMirror active sync 1
ONTAP Mediator overview 1
What’s new with the ONTAP Mediator 2
Install or upgrade 5
Manage the ONTAP mediator service 46
Maintain OS host for ONTAP Mediator 60

Mediator service for MetroCluster and SnapMirror active sync

ONTAP Mediator overview

The ONTAP Mediator provides several functions for ONTAP features:

- Provides a persistent and fenced store for HA metadata.
- Serves as a ping proxy for controller liveliness.
- Provides synchronous node health query functionality to aid in quorum determination.

The ONTAP Mediator provides two additional systemctl services:

- **`ontap_mediator.service`**

  Maintains the REST APIs server for managing the ONAP 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.

Special Notes

ONTAP Mediator relies on SCST to provide iSCSI (See http://scst.sourceforge.net/index.html). This package is a kernel module that is compiled during installation specifically for the kernel. Any updates to the kernel might require SCST to be re-installed. Alternatively, uninstall then re-install the ONTAP Mediator, then reconfigure the ONTAP relationship.

Any updates to the server OS kernel should be coordinated with a maintenance window in ONTAP.

What's new with the ONTAP Mediator

New enhancements to the ONTAP Mediator are provided with each release. Here's what's new.

Enhancements

ONTAP Mediator version	Enhancements
1.7	• Support for RHEL 8.5, 8.6, 8.7, 8.8, 8.9, 9.0, 9.1, 9.2, and 9.3 • Support for Rocky Linux 8 and 9
1.6	• Python 3.9 updates. • Support for RHEL 8.4-8.8, 9.0-9.2, Rocky Linux 8 and 9. • Discontinued support for RHEL 7.x / CentOS all releases.
1.5	• Optimizes speed for larger scale SnapMirror active sync systems. • Cryptographic code-signature added to the installer. • Includes deprecation warnings for RHEL 7.x / CentOS 7.x.
1.4	• Support for RHEL 8.4 and 8.5. • Includes SCST version 3.6.0. • Added support for UFEI-based firmware's Secure Boot (SB).
1.3	• Support for RHEL/CentOS 8.2 and 8.3. • Includes SCST version 3.5.0.
1.2	• Support for HTTPs mailboxes. • For use with ONTAP 9.8+ MCC-IP AUSO and SnapMirror active sync ZRTO. • Includes SCST version 3.4.0.

1.1	• Support for RHEL/CentOS 7.6, 7.7, 8.0, and 8.1. • Eliminates Perl dependencies. • Includes SCST version 3.4.0.
1.0	• Support for iSCSI mailboxes. • For use with ONTAP 9.7+ MCC-IP AUSO. • Support for RHEL/CentOS 7.6.

OS support matrix

OS for ONTAP Mediator	1.7	1.6	1.5	1.4	1.3	1.2	1.1	1.0
7.6	Obsolete	Obsolete	Yes	Yes	Yes	Yes	Yes	Yes (RHEL only)
7.7	Obsolete	Obsolete	Yes	Yes	Yes	Yes	No	No
7.8	Obsolete	Obsolete	Yes	Yes	Yes	Yes	No	No
7.9	Obsolete	Obsolete	Yes	Yes	Yes	Implied	No	No
RHEL 8.0	Obsolete	Obsolete	Yes	Yes	Yes	Yes	Yes	No
RHEL 8.1	Obsolete	Obsolete	Yes	Yes	Yes	Yes	No	No
RHEL 8.2	Obsolete	Obsolete	Yes	Yes	Yes	No	No	No
RHEL 8.3	Obsolete	Obsolete	Yes	Yes	Yes	No	No	No
RHEL 8.4	Obsolete	Yes	Yes	Yes	No	No	No	No
RHEL 8.5	Yes	Yes	Yes	Yes	No	No	No	No
RHEL 8.6	Yes	Yes	No	No	No	No	No	No
RHEL 8.7	Yes	Yes	No	No	No	No	No	No
RHEL 8.8	Yes	Yes	No	No	No	No	No	No
RHEL 9.0	Yes	Yes	No	No	No	No	No	No

RHEL 9.1	Yes	Yes	No	No	No	No	No	No
RHEL 9.2	Yes	Yes	No	No	No	No	No	No
RHEL 9.3	Yes	No	No	No	No	No	No	No
CentOS 8 and stream	No	No	No	No	No	N/A	N/A	N/A
Rocky Linux 8	Yes	Yes	N/A	N/A	N/A	N/A	N/A	N/A
Rocky Linux 9	Yes	Yes	N/A	N/A	N/A	N/A	N/A	N/A

- OS refers to both RedHat and CentOS releases unless otherwise specified.
- "No" means that the OS and ONTAP Mediator are not compatible.
- 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.
- ONTAP Mediator 1.5 was the last supported release for RHEL 7.x branch operating systems.
- ONTAP Mediator 1.6 adds support for Rocky Linux 8 and 9.

Resolved issues

Date of change	Change ID	Description
10 Jan 2023	6567145	The following changes were made: • Added support for additional operating systems for ONTAP Mediator: RHEL 9.6, 8.7, 9.0, and 9.1. • Added new SCST version 3.7.0 to unblock issues for newly supported operating systems. • Added support for Rocky Linux: Rocky 8 and 9.
24 Jan 2023	6621319	Allowed pre-installed SCST library for ONTAP Mediator installations.
27 Feb 2023	6623764	Implemented changes to always load the scst_disk kernel module when the mediator-scst service restarts. These changes ensure the service will always be ready to create new iSCSI targets using the standard logic.
28 Feb 2023	6625194	Added a new option to the ONTAP Mediator installer: `--skip -yum-dependencies`

24 Mar 2023	6652840	Updated the ONTAP Mediator installer so that it is able to reinstall or repair the SCST installation.
27 Mar 2023	6655179	Fixed a parsing issue that occurred when the support bundle collection with a complex password was triggered.
28 Mar 2023	6656739	Changed the SCST comparison logic so that is will install the right version when ONTAP Mediator is upgraded.

Install or upgrade

Prepare to install or upgrade the ONTAP Mediator service

To install the ONTAP Mediator service, you must ensure all prerequisites are met, fetch the installation package and run the installer on the host. This procedure is used for an installation or an upgrade of an existing installation.

About this task

- Beginning with ONTAP 9.7, you can use any version of ONTAP Mediator to monitor a MetroCluster IP configuration.
- Beginning with ONTAP 9.8, you can use any version of ONTAP Mediator to monitor an SnapMirror active sync relationship.

Before you begin

You must meet the following prerequisites.

ONTAP Mediator version	Supported Linux versions
1.7	• Red Hat Enterprise Linux: 8.5, 8.6, 8.7, 8.8, 8.9, 9.0, 9.1, 9.2, and 9.3 • Rocky Linux 8 and 9
1.6	• Red Hat Enterprise Linux: 8.4, 8.5, 8.6, 8.7, 8.8, 9.0, 9.1, 9.2 • Rocky Linux 8 and 9
1.5	• Red Hat Enterprise Linux: 7.6, 7.7, 7.8, 7.9, 8.1, 8.2, 8.3, 8.4, 8.5 • CentOS: 7.6, 7.7, 7.8, 7.9
1.4	• Red Hat Enterprise Linux: 7.6, 7.7, 7.8, 7.9, 8.1, 8.2, 8.3, 8.4, 8.5 • CentOS: 7.6, 7.7, 7.8, 7.9
1.3	• Red Hat Enterprise Linux: 7.6, 7.7, 7.8, 7.9, 8.1, 8.2, 8.3 • CentOS: 7.6, 7.7, 7.8, 7.9

1.2	• Red Hat Enterprise Linux: 7.6, 7.7, 7.8, 8.1 • CentOS: 7.6, 7.7, 7.8

The kernel version must match the operating system version.

- 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

Any library packages except the kernel can safely be updated but might require a reboot to take affect within the ONTAP Mediator application. A service window is recommended when a reboot is required.

If you install the `yum-utils` package, you can use the `needs-restarting` command.

The kernel core can be updated if it is being updated to a version that is still supported by the ONTAP Mediator version matrix. A reboot will be mandatory, so a service window is required.

The SCST kernel module must be uninstalled prior to the reboot, then re-installed after the reboot.

Upgrading to a kernel beyond the supported OS release for the specific ONTAP Mediator release is not support. (This likely indicates that the tested SCST module won't compile).

Register a security key when UEFI Secure Boot is enabled

If UEFI Secure Boot is enabled, to install ONTAP Mediator, you will have to register a security key before the ONTAP Mediator service can start. To determine if the system is UEFI-enabled and Secure Boot is turned on, perform the following steps:

Steps

1. If mokutil is not installed, run the following command:

   ```
   yum install mokutil
   ```

2. To determine if UEFI Secure Boot is enabled on your system, run the following command:

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

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

   ONTAP Mediator 1.2.0 and previous versions do not support this mode.

Disable UEFI Secure Boot

You can also choose to disable UEFI Secure Boot before installing ONTAP Mediator.

Steps

1. In the physical machine BIOS settings, disable the "UEFI Secure Boot" option.

2. In the VMware settings for the VM, disable the "Safe Start" option for vSphere 6.x or the "Secure Boot" option for vSphere 7.x.

Upgrade the host operating system and then the ONTAP Mediator

To upgrade the host OS for ONTAP Mediator to a later version, you must first uninstall ONTAP Mediator.

Before you begin

The best practices for installing Red Hat Enterprise Linux or Rocky Linux and the associated repositories on your system are listed below. Systems installed or configured differently might require additional steps.

- You must install Red Hat Enterprise Linux or Rocky Linux according to Red Hat best practices. Due to end-of-life support for CentOS 8.x versions, compatible versions of CentOS 8.x are not recommended.
- While installing the ONTAP Mediator service on Red Hat Enterprise Linux or Rocky Linux, the system must have access to the appropriate repository so that the installation program can access and install all the required software dependencies.
- For the yum installer to find dependent software in the Red Hat Enterprise Linux repositories, you must have registered the system during the Red Hat Enterprise Linux installation or afterwards by using a valid Red Hat subscription.

  See the Red Hat documentation for information about the Red Hat Subscription Manager.

- The following ports must be unused and available for the Mediator:
  - 31784
  - 3260
- If using a third-party firewall: refer to Firewall requirements for ONTAP Mediator
- If the Linux host is in a location without access to the internet, you must ensure that the required packages are available in a local repository.

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

What you'll need

The following packages are required by the ONTAP Mediator service:

All RHEL/CentOS versions	Additional packages for RHEL 8.x / Rocky Linux 8	Additional packages for RHEL 9.x / Rocky Linux 9

• openssl • openssl-devel • kernel-devel-$ (uname -r) • gcc • make • libselinux-utils • patch • bzip2 • perl-Data-Dumper • perl-ExtUtils-MakeMaker • efibootmgr • mokutil	• python3-pip • elfutils-libelf-devel • policycoreutils-python-utils • redhat-lsb-core • python39 • python39-devel	• python3-pip • elfutils-libelf-devel • policycoreutils-python-utils • python3 • python3-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 certification is recommended.

About this task

When you upgrade the host OS for ONTAP Mediator to a later major version (for example, from 7.x to 8.x) using the leapp-upgrade tool, you must uninstall ONTAP Mediator because the tool tries to detect new versions of any RPMs that are installed in the repositories that are registered with the system.

Because an .rpm file was installed as part of the ONTAP Mediator installer, it is included in that search. However, because that .rpm file was unpacked as part of the installer and not downloaded from a registered repository, an upgrade cannot be found. In this case, the leapp-upgrade tool uninstalls the package.

In order to preserve the log files, which will be used to triage support cases, you should back up the files prior to doing an OS upgrade and restore them after a reinstall of the ONTAP Mediator package. Because the ONTAP Mediator is being reinstalled, any ONTAP Clusters that are connected to it will need to be reconnected after the new installation.

The following steps should be performed in order. Immediately after you reinstall ONTAP Mediator, you should stop the ontap_mediator service, replace the log files, and restart the service. This will ensure logs will not be lost.

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 8.4
  ..<snip upgrade checks>..
  ..<fix issues found>..
[rootmediator-host ~]# leapp upgrade --target 8.4
  ..<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-1.6.0/ontap-mediator-1.6.0

ONTAP Mediator: Self Extracting Installer

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

4. Stop the ontap_mediator service.

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

5. Replace the log files.

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

6. Start the ontap_mediator service.

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

7. Reconnect all ONTAP clusters to the upgraded ONTAP Mediator

Procedure for 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::>
```

Procedure for SnapMirror active sync

For SnapMirror active sync, if you installed your TLS certificate outside of the /opt/netapp directory, then you will not need to reinstall it. If you were using the default generated self-signed certificate or put your custom certificate in the /opt/netapp directory, then you should back it up and restore it.

```
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::>
```

Enable access to the repositories

You should enable access to repositories so 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…
RHEL 7.x	• rhel-7-server-optional-rpms
RHEL 8.x	• rhel-8-for-x86_64-baseos-rpms • rhel-8-for-x86_64-appstream-rpms
RHEL 9.x	• rhel-9-for-x86_64-baseos-rpms • rhel-9-for-x86_64-appstream-rpms
CentOS 7.x	• C7.6.1810 - Base repository
Rocky Linux 8	• appstream • baseos
Rocky Linux 9	• 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.

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 `yum repolist` command.

   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 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 `yum repolist` command.

   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 `yum repolist` command.

   The newly subscribed repositories should appear in the list.

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 `yum repolist` command.

   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 8 or 9 operating systems

Use this procedure if your operating system is **Rocky Linux 8** or **Rocky Linux 9** 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 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 ~]#
```

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 ~]#
```

Download the Mediator installation package

Download the Mediator installation package as part of the installation process.

Steps

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

   ONTAP Mediator download page

2. Confirm that the Mediator installation package is in the current working directory:

   `ls`

   ```
   [root@mediator-host ~]#ls
   ontap-mediator-1.7.0.tgz
   ```

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

   If you are at a location without access to the internet, you must ensure that the installer has access to the required packages.

3. If necessary, move the Mediator installation package from the download directory to the installation directory on the Linux Mediator host.
4. Unzip the installer package:

   `tar xvfz ontap-mediator-1.7.0.tgz`

   ```
   [root@scs000099753 ~]# tar xvfz ontap-mediator-1.7.0.tgz
   ontap-mediator-1.7.0/
   ontap-mediator-1.7.0/ONTAP-Mediator-production.pub
   ontap-mediator-1.7.0/tsa-prod-chain-ONTAP-Mediator.pem
   ontap-mediator-1.7.0/tsa-prod-ONTAP-Mediator.pem
   ontap-mediator-1.7.0/csc-prod-ONTAP-Mediator.pem
   ontap-mediator-1.7.0/csc-prod-chain-ONTAP-Mediator.pem
   ontap-mediator-1.7.0/ontap-mediator-1.7.0
   ontap-mediator-1.7.0/ontap-mediator-1.7.0.sig.tsr
   ontap-mediator-1.7.0/ontap-mediator-1.7.0.tsr
   ontap-mediator-1.7.0/ontap-mediator-1.7.0.sig
   ```

Verify the ONTAP Mediator code signature

You should verify the ONTAP Mediator code signature before installing the Mediator installation package.

Before you begin

Before verifying the Mediator code signature, your system must meet the following requirements.

- 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 following files are included in the download package:

File	Description
`ONTAP-Mediator-development.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-1.7.0`	The product installation executable for version 1.7.0
`ontap-mediator-1.7.0.sig`	The SHA-256 hashed, then RSA-signed using the csc-prod key, signature for the installer
`ontap-mediator-1.7.0.sig.tsr`	The revocation request for use by OCSCP for the installer's signature
`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 used to register the certificate because 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  # Failure action: The Code-
   Signature-Check certificate has expired or is invalid. Download a newer
   version of the ONTAP Mediator.
   openssl verify -untrusted tsa-prod-chain-ONTAP-Mediator.pem -CApath
   ${OPENSSLDIR} tsa-prod-ONTAP-Mediator.pem  # Failure action: The Time-
   Stamp certificate has expired or is invalid. Download a newer version of
   the ONTAP Mediator.
   ```

3. Verify the `ontap-mediator-1.6.0.sig.tsr` and `ontap-mediator-1.7.0.tsr` files using the associated certificates:

   `openssl ts -verify`

   > `.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. The verification is performed locally on your machine. Independently, there is no need to access TSA servers.

   ```
   openssl ts -verify -data ontap-mediator-1.7.0.sig -in ontap-mediator-
   1.7.0.sig.tsr -CAfile tsa-prod-chain-ONTAP-Mediator.pem -untrusted tsa-
   prod-ONTAP-Mediator.pem
   openssl ts -verify -data ontap-mediator-1.7.0 -in ontap-mediator-
   1.7.0.tsr -CAfile tsa-prod-chain-ONTAP-Mediator.pem -untrusted tsa-prod-
   ONTAP-Mediator.pem
   ```

4. Verify signatures against the key:

   `openssl -dgst -verify`

   ```
   openssl dgst -sha256 -verify ONTAP-Mediator-production.pub -signature
   ontap-mediator-1.7.0.sig ontap-mediator-1.7.0
   ```

Example of Verifying the ONTAP Mediator code signature (console output)

```
[root@scspa2695423001 ontap-mediator-1.7.0]# pwd
/root/ontap-mediator-1.7.0
[root@scspa2695423001 ontap-mediator-1.7.0]# ls -l
total 63660
-r--r--r-- 1 root root     8582 Feb 19 15:02 csc-prod-chain-ONTAP-
Mediator.pem
-r--r--r-- 1 root root     2373 Feb 19 15:02 csc-prod-ONTAP-
Mediator.pem
-r-xr-xr-- 1 root root 65132818 Feb 20 15:17 ontap-mediator-1.7.0
-rw-r--r-- 1 root root      384 Feb 20 15:17 ontap-mediator-1.7.0.sig
-rw-r--r-- 1 root root     5437 Feb 20 15:17 ontap-mediator-
1.7.0.sig.tsr
-rw-r--r-- 1 root root     5436 Feb 20 15:17 ontap-mediator-1.7.0.tsr
-r--r--r-- 1 root root      625 Feb 19 15:02 ONTAP-Mediator-
production.pub
-r--r--r-- 1 root root     3323 Feb 19 15:02 tsa-prod-chain-ONTAP-
Mediator.pem
-r--r--r-- 1 root root     1740 Feb 19 15:02 tsa-prod-ONTAP-
Mediator.pem
[root@scspa2695423001 ontap-mediator-1.7.0]#
[root@scspa2695423001 ontap-mediator-1.7.0]#
/root/verify_ontap_mediator_signatures.sh
++ openssl version -d
++ cut -d '"' -f2
+ OPENSSLDIR=/etc/pki/tls
+ openssl version
OpenSSL 1.1.1k  FIPS 25 Mar 2021
++ openssl x509 -noout -ocsp_uri -in csc-prod-chain-ONTAP-Mediator.pem
+ ocsp_uri=http://ocsp.entrust.net
+ echo http://ocsp.entrust.net
http://ocsp.entrust.net
+ 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
+ openssl ocsp -issuer csc-prod-chain-ONTAP-Mediator.pem -CAfile csc-
prod-chain-ONTAP-Mediator.pem -cert csc-prod-ONTAP-Mediator.pem -url
http://ocsp.entrust.net -resp_text -respout resp.der -verify_other csc-
prod-chain-ONTAP-Mediator.pem
OCSP Response Data:
    OCSP Response Status: successful (0x0)
    Response Type: Basic OCSP Response
    Version: 1 (0x0)
    Responder Id: C = US, O = "Entrust, Inc.", CN = Entrust Extended
Validation Code Signing CA - EVCS2
```

```
    Produced At: Feb 28 05:01:00 2023 GMT
    Responses:
    Certificate ID:
      Hash Algorithm: sha1
      Issuer Name Hash: 69FA640329AB84E27220FE0927647B8194B91F2A
      Issuer Key Hash: CE894F8251AA15A28462CA312361D261FBF8FE78
      Serial Number: 511A542B57522AEB7295A640DC6200E5
    Cert Status: good
    This Update: Feb 28 05:00:00 2023 GMT
    Next Update: Mar  4 04:59:59 2023 GMT

    Signature Algorithm: sha512WithRSAEncryption
         3c:1d:49:b0:93:62:37:3e:c7:38:e3:9f:9f:62:82:73:ed:f4:
         ea:00:6b:f1:01:cd:79:57:92:f1:9d:5d:85:9b:60:59:f8:6c:
         e6:f4:50:51:f3:4c:8a:51:dd:50:68:16:8f:20:24:7e:39:b0:
         44:94:8d:b0:61:da:b9:08:36:74:2d:44:55:62:fb:92:be:4a:
         e7:6c:8c:49:dd:0c:fd:d8:ce:20:08:0d:0f:5a:29:a3:19:03:
         9f:d3:df:41:f4:89:0f:73:18:3f:ac:bb:a7:a3:96:7d:c5:70:
         4c:57:cd:17:17:c6:8a:60:d1:37:c9:2d:81:07:2a:d7:a6:02:
         ee:ce:88:16:22:db:e3:43:64:1e:9b:0d:4d:31:66:fa:ab:a5:
         52:99:94:4a:4a:d0:52:c5:34:f5:18:c7:15:5b:ce:74:c2:fc:
         61:ea:55:aa:f1:2f:82:a3:6a:95:8d:7e:2b:38:49:4f:bf:b1:
         68:7b:1b:24:8b:1f:4d:c5:77:f0:71:af:9c:34:c8:7a:82:50:
         09:a2:19:6e:c6:30:4f:da:a2:79:08:f9:d0:ff:85:d9:2a:84:
         cf:0c:aa:75:8f:72:c9:a7:a2:83:e8:8b:cf:ed:0c:69:75:b6:
         2a:7b:6b:58:99:01:d8:34:ad:e1:89:25:27:1b:fa:d9:6d:32:
         97:3a:0b:0a:8e:a3:9e:e3:f4:e0:d6:1a:c9:b5:14:8c:3e:54:
         3b:37:17:1a:93:44:84:8b:4a:87:97:1e:76:43:3e:d3:ec:8b:
         7e:56:4a:3f:01:31:c0:e5:58:fb:50:ce:6f:b1:e7:35:f9:b7:
         a3:ef:6b:3b:21:95:37:a6:5b:8f:f0:15:18:36:65:89:a1:9c:
         9b:69:00:b4:b1:65:6a:bc:11:2d:d4:9b:b4:97:cc:cb:7a:0c:
         16:11:c1:75:58:7e:13:ab:56:3c:3f:93:5b:95:24:c6:54:52:
         1f:86:a9:16:ce:d9:ea:8b:3a:f3:4f:c4:8f:ad:de:e8:3e:3c:
         d2:51:51:ad:33:7f:d8:c5:33:24:26:f1:2d:9d:0e:9f:55:d0:
         68:bf:af:bd:68:4a:40:08:bc:92:a0:62:54:7d:16:7b:36:29:
         15:b1:cd:58:8e:fb:4a:f2:3e:94:8b:fe:56:95:cc:24:32:af:
         5f:71:99:18:ed:0c:64:94:f7:54:48:87:48:d0:6d:b3:42:04:
         96:03:73:a2:8e:8a:6a:b2:af:ee:56:19:a1:c6:35:12:59:ad:
         19:6a:fe:e0:f1:27:cc:96:4e:f0:4f:fb:6a:bd:ce:05:2c:aa:
         79:7c:df:02:5c:ca:53:7d:60:12:88:7c:ce:15:c7:d4:02:27:
         c1:ab:cf:71:30:1e:14:ba
WARNING: no nonce in response
Response verify OK
csc-prod-ONTAP-Mediator.pem: good
        This Update: Feb 28 05:00:00 2023 GMT
        Next Update: Mar  4 04:59:59 2023 GMT
```

```
+ openssl verify -untrusted csc-prod-chain-ONTAP-Mediator.pem -CApath
/etc/pki/tls csc-prod-ONTAP-Mediator.pem
csc-prod-ONTAP-Mediator.pem: OK
+ openssl verify -untrusted tsa-prod-chain-ONTAP-Mediator.pem -CApath
/etc/pki/tls tsa-prod-ONTAP-Mediator.pem
tsa-prod-ONTAP-Mediator.pem: OK
+ openssl ts -verify -data ontap-mediator-1.7.0.sig -in ontap-mediator-
1.7.0.sig.tsr -CAfile tsa-prod-chain-ONTAP-Mediator.pem -untrusted tsa-
prod-ONTAP-Mediator.pem
Using configuration from /etc/pki/tls/openssl.cnf
Verification: OK
+ openssl ts -verify -data ontap-mediator-1.7.0 -in ontap-mediator-
1.7.0.tsr -CAfile tsa-prod-chain-ONTAP-Mediator.pem -untrusted tsa-
prod-ONTAP-Mediator.pem
Using configuration from /etc/pki/tls/openssl.cnf
Verification: OK
+ openssl dgst -sha256 -verify ONTAP-Mediator-production.pub -signature
ontap-mediator-1.7.0.sig ontap-mediator-1.7.0
Verified OK
[root@scspa2695423001 ontap-mediator-1.7.0]#
```

Install the ONTAP Mediator installation package

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

Steps

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

   `./ontap-mediator-1.7.0/ontap-mediator-1.7.0 -y`

   ```
   [root@scs000099753 ~]# ./ontap-mediator-1.5.0/ontap-mediator-1.7.0 -y
   ```

   The installation process proceeds to create the required accounts and install required packages. If you have a previous version of Mediator installed on the host, you will be prompted to confirm that you want to upgrade.

2. 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:

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

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

   - Locate the required keys:

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

After installation, the README files and key location are also provided in the system output.

Example of ONTAP Mediator 1.6 installation (console output)

```
[root@scs000099753 ~]# ./ontap-mediator-1.6.0/ontap-mediator-1.6.0 -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

+ 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
success
success
success


##################################################################
Preparing for installation of ONTAP Mediator packages.


+ Installing required packages.


Last metadata expiration check: 0:25:24 ago on Fri 21 Oct 2022 04:00:13
PM EDT.
Package openssl-1:1.1.1k-4.el8.x86_64 is already installed.
Package gcc-8.4.1-1.el8.x86_64 is already installed.
Package python36-3.6.8-2.module+el8.1.0+3334+5cb623d7.x86_64 is already
installed.
Package libselinux-utils-2.9-5.el8.x86_64 is already installed.
Package perl-Data-Dumper-2.167-399.el8.x86_64 is already installed.
Package efibootmgr-16-1.el8.x86_64 is already installed.
Package mokutil-1:0.3.0-11.el8.x86_64 is already installed.
```

```
Package python3-pip-9.0.3-19.el8.noarch is already installed.
Package policycoreutils-python-utils-2.9-14.el8.noarch is already
installed.
Dependencies resolved.
================================================================================
================================================================================
 Package                                    Architecture
Version                                                 Repository
Size
================================================================================
================================================================================
Installing:
 bzip2                                      x86_64
1.0.6-26.el8                                            rhel-8-for-
x86_64-baseos-rpms                          60 k
 elfutils-libelf-devel                      x86_64
0.186-1.el8                                             rhel-8-for-
x86_64-baseos-rpms                          60 k
 kernel-devel                               x86_64
4.18.0-348.el8                                          rhel-8-for-
x86_64-baseos-rpms                          20 M
 make                                       x86_64
1:4.2.1-11.el8                                          rhel-8-for-
x86_64-baseos-rpms                         498 k
 openssl-devel                              x86_64
1:1.1.1k-7.el8_6                                        rhel-8-for-
x86_64-baseos-rpms                         2.3 M
 patch                                      x86_64
2.7.6-11.el8                                            rhel-8-for-
x86_64-baseos-rpms                         138 k
 perl-ExtUtils-MakeMaker                    noarch
1:7.34-1.el8                                            rhel-8-for-
x86_64-appstream-rpms                      301 k
 python36-devel                             x86_64
3.6.8-38.module+el8.5.0+12207+5c5719bc                  rhel-8-for-
x86_64-appstream-rpms                       17 k
 redhat-lsb-core                            x86_64
4.1-47.el8                                              rhel-8-for-
x86_64-appstream-rpms                       45 k
Upgrading:
 cpp                                        x86_64
8.5.0-10.1.el8_6                                        rhel-8-for-
x86_64-appstream-rpms                       10 M
 elfutils-libelf                            x86_64
```

```
0.186-1.el8                                                    rhel-8-for-
x86_64-baseos-rpms                        229 k
 elfutils-libs                                 x86_64
0.186-1.el8                                                    rhel-8-for-
x86_64-baseos-rpms                        295 k
 gcc                                           x86_64
8.5.0-10.1.el8_6                                               rhel-8-for-
x86_64-appstream-rpms                      23 M
 libgcc                                        x86_64
8.5.0-10.1.el8_6                                               rhel-8-for-
x86_64-baseos-rpms                         80 k
 libgomp                                       x86_64
8.5.0-10.1.el8_6                                               rhel-8-for-
x86_64-baseos-rpms                        207 k
 libsemanage                                   x86_64
2.9-8.el8                                                      rhel-8-for-
x86_64-baseos-rpms                        168 k
 mokutil                                       x86_64
1:0.3.0-11.el8_6.1                                             rhel-8-for-
x86_64-baseos-rpms                         46 k
 openssl                                       x86_64
1:1.1.1k-7.el8_6                                               rhel-8-for-
x86_64-baseos-rpms                        709 k
 openssl-libs                                  x86_64
1:1.1.1k-7.el8_6                                               rhel-8-for-
x86_64-baseos-rpms                        1.5 M
 platform-python-pip                           noarch
9.0.3-22.el8                                                   rhel-8-for-
x86_64-baseos-rpms                        1.6 M
 policycoreutils                               x86_64
2.9-19.el8                                                     rhel-8-for-
x86_64-baseos-rpms                        374 k
 policycoreutils-python-utils                  noarch
2.9-19.el8                                                     rhel-8-for-
x86_64-baseos-rpms                        253 k
 python3-libsemanage                           x86_64
2.9-8.el8                                                      rhel-8-for-
x86_64-baseos-rpms                        128 k
 python3-pip                                   noarch
9.0.3-22.el8                                                   rhel-8-for-
x86_64-appstream-rpms                      20 k
 python3-policycoreutils                       noarch
2.9-19.el8                                                     rhel-8-for-
x86_64-baseos-rpms                        2.2 M
 python36                                      x86_64
3.6.8-38.module+el8.5.0+12207+5c5719bc                         rhel-8-for-
```

```
x86_64-appstream-rpms                    19 k
Installing dependencies:
 annobin                                 x86_64
10.29-3.el8                                        rhel-8-for-
x86_64-appstream-rpms                   117 k
 at                                      x86_64
3.1.20-11.el8                                      rhel-8-for-
x86_64-baseos-rpms                       81 k
 bc                                      x86_64
1.07.1-5.el8                                       rhel-8-for-
x86_64-baseos-rpms                      129 k
 cups-client                             x86_64
1:2.2.6-38.el8                                     rhel-8-for-
x86_64-appstream-rpms                   169 k
 dwz                                     x86_64
0.12-10.el8                                        rhel-8-for-
x86_64-appstream-rpms                   109 k
 ed                                      x86_64
1.14.2-4.el8                                       rhel-8-for-
x86_64-baseos-rpms                       82 k
 efi-srpm-macros                         noarch
3-3.el8                                            rhel-8-for-
x86_64-appstream-rpms                    22 k
 esmtp                                   x86_64
1.2-15.el8                                         EPEL-8
57 k
 ghc-srpm-macros                         noarch
1.4.2-7.el8                                        rhel-8-for-
x86_64-appstream-rpms                   9.4 k
 go-srpm-macros                          noarch
2-17.el8                                           rhel-8-for-
x86_64-appstream-rpms                    13 k
 keyutils-libs-devel                     x86_64
1.5.10-6.el8                                       rhel-8-for-
x86_64-baseos-rpms                       48 k
 krb5-devel                              x86_64
1.18.2-14.el8                                      rhel-8-for-
x86_64-baseos-rpms                      560 k
 libcom_err-devel                        x86_64
1.45.6-2.el8                                       rhel-8-for-
x86_64-baseos-rpms                       38 k
 libesmtp                                x86_64
1.0.6-18.el8                                       EPEL-8
70 k
 libkadm5                                x86_64
1.18.2-14.el8                                      rhel-8-for-
```

```
x86_64-baseos-rpms                      187 k
 liblockfile                                    x86_64
1.14-1.el8                                              rhel-8-for-
x86_64-appstream-rpms                    32 k
 libselinux-devel                               x86_64
2.9-5.el8                                               rhel-8-for-
x86_64-baseos-rpms                      200 k
 libsepol-devel                                 x86_64
2.9-3.el8                                               rhel-8-for-
x86_64-baseos-rpms                       87 k
 libverto-devel                                 x86_64
0.3.0-5.el8                                             rhel-8-for-
x86_64-baseos-rpms                       18 k
 m4                                             x86_64
1.4.18-7.el8                                            rhel-8-for-
x86_64-baseos-rpms                      223 k
 mailx                                          x86_64
12.5-29.el8                                             rhel-8-for-
x86_64-baseos-rpms                      257 k
 ncurses-compat-libs                            x86_64
6.1-9.20180224.el8                                      rhel-8-for-
x86_64-baseos-rpms                      328 k
 ocaml-srpm-macros                              noarch
5-4.el8                                                 rhel-8-for-
x86_64-appstream-rpms                   9.5 k
 openblas-srpm-macros                           noarch
2-2.el8                                                 rhel-8-for-
x86_64-appstream-rpms                   8.0 k
 pcre2-devel                                    x86_64
10.32-2.el8                                             rhel-8-for-
x86_64-baseos-rpms                      605 k
 pcre2-utf16                                    x86_64
10.32-2.el8                                             rhel-8-for-
x86_64-baseos-rpms                      229 k
 pcre2-utf32                                    x86_64
10.32-2.el8                                             rhel-8-for-
x86_64-baseos-rpms                      220 k
 perl-CPAN-Meta-YAML                            noarch
0.018-397.el8                                           rhel-8-for-
x86_64-appstream-rpms                    34 k
 perl-ExtUtils-Command                          noarch
1:7.34-1.el8                                            rhel-8-for-
x86_64-appstream-rpms                    19 k
 perl-ExtUtils-Install                          noarch
2.14-4.el8                                              rhel-8-for-
x86_64-appstream-rpms                    46 k
```

```
 perl-ExtUtils-Manifest                          noarch
1.70-395.el8                                             rhel-8-for-
x86_64-appstream-rpms                    37 k
 perl-ExtUtils-ParseXS                           noarch
1:3.35-2.el8                                             rhel-8-for-
x86_64-appstream-rpms                    83 k
 perl-JSON-PP                                    noarch
1:2.97.001-3.el8                                         rhel-8-for-
x86_64-appstream-rpms                    68 k
 perl-Math-BigInt                                noarch
1:1.9998.11-7.el8                                        rhel-8-for-
x86_64-baseos-rpms                      196 k
 perl-Math-Complex                               noarch
1.59-421.el8                                             rhel-8-for-
x86_64-baseos-rpms                      109 k
 perl-Test-Harness                               noarch
1:3.42-1.el8                                             rhel-8-for-
x86_64-appstream-rpms                   279 k
 perl-devel                                      x86_64
4:5.26.3-419.el8_4.1                                     rhel-8-for-
x86_64-appstream-rpms                   599 k
 perl-srpm-macros                                noarch
1-25.el8                                                 rhel-8-for-
x86_64-appstream-rpms                    11 k
 perl-version                                    x86_64
6:0.99.24-1.el8                                          rhel-8-for-
x86_64-appstream-rpms                    67 k
 platform-python-devel                           x86_64
3.6.8-41.el8                                             rhel-8-for-
x86_64-appstream-rpms                   249 k
 python-rpm-macros                               noarch
3-41.el8                                                 rhel-8-for-
x86_64-appstream-rpms                    15 k
 python-srpm-macros                              noarch
3-41.el8                                                 rhel-8-for-
x86_64-appstream-rpms                    15 k
 python3-pyparsing                               noarch
2.1.10-7.el8                                             rhel-8-for-
x86_64-baseos-rpms                      142 k
 python3-rpm-generators                          noarch
5-7.el8                                                  rhel-8-for-
x86_64-appstream-rpms                    25 k
 python3-rpm-macros                              noarch
3-41.el8                                                 rhel-8-for-
x86_64-appstream-rpms                    14 k
 qt5-srpm-macros                                 noarch
```

```
5.15.2-1.el8                                                rhel-8-for-
x86_64-appstream-rpms                    11 k
 redhat-lsb-submod-security                      x86_64
4.1-47.el8                                                  rhel-8-for-
x86_64-appstream-rpms                    22 k
 redhat-rpm-config                               noarch
125-1.el8                                                   rhel-8-for-
x86_64-appstream-rpms                    87 k
 rust-srpm-macros                                noarch
5-2.el8                                                     rhel-8-for-
x86_64-appstream-rpms                   9.3 k
 spax                                            x86_64
1.5.3-13.el8                                                rhel-8-for-
x86_64-baseos-rpms                      217 k
 systemtap-sdt-devel                             x86_64
4.6-4.el8                                                   rhel-8-for-
x86_64-appstream-rpms                    86 k
 time                                            x86_64
1.9-3.el8                                                   rhel-8-for-
x86_64-baseos-rpms                       54 k
 unzip                                           x86_64
6.0-46.el8                                                  rhel-8-for-
x86_64-baseos-rpms                      196 k
 util-linux-user                                 x86_64
2.32.1-28.el8                                               rhel-8-for-
x86_64-baseos-rpms                      100 k
 zip                                             x86_64
3.0-23.el8                                                  rhel-8-for-
x86_64-baseos-rpms                      270 k
 zlib-devel                                      x86_64
1.2.11-17.el8                                               rhel-8-for-
x86_64-baseos-rpms                       58 k
Installing weak dependencies:
 perl-CPAN-Meta                                  noarch
2.150010-396.el8                                            rhel-8-for-
x86_64-appstream-rpms                   191 k
 perl-CPAN-Meta-Requirements                     noarch
2.140-396.el8                                               rhel-8-for-
x86_64-appstream-rpms                    37 k
 perl-Encode-Locale                              noarch
1.05-10.module+el8.3.0+6498+9eecfe51                        rhel-8-for-
x86_64-appstream-rpms                    22 k
 perl-Time-HiRes                                 x86_64
4:1.9758-2.el8                                              rhel-8-for-
x86_64-appstream-rpms                    61 k
```

```
Transaction Summary
================================================================================
================================================================================
=========================================
Install  69 Packages
Upgrade  17 Packages

Total download size: 72 M
Is this ok [y/N]: y
Downloading Packages:
(1/86): perl-ExtUtils-Install-2.14-4.el8.noarch.rpm
735 kB/s |  46 kB     00:00
(2/86): libesmtp-1.0.6-18.el8.x86_64.rpm
1.0 MB/s |  70 kB     00:00
(3/86): esmtp-1.2-15.el8.x86_64.rpm
747 kB/s |  57 kB     00:00
(4/86): rust-srpm-macros-5-2.el8.noarch.rpm
308 kB/s | 9.3 kB     00:00
(5/86): perl-ExtUtils-Manifest-1.70-395.el8.noarch.rpm
781 kB/s |  37 kB     00:00
(6/86): perl-CPAN-Meta-2.150010-396.el8.noarch.rpm
2.7 MB/s | 191 kB     00:00
(7/86): ocaml-srpm-macros-5-4.el8.noarch.rpm
214 kB/s | 9.5 kB     00:00
(8/86): perl-JSON-PP-2.97.001-3.el8.noarch.rpm
1.2 MB/s |  68 kB     00:00
(9/86): perl-ExtUtils-MakeMaker-7.34-1.el8.noarch.rpm
5.8 MB/s | 301 kB     00:00
(10/86): ghc-srpm-macros-1.4.2-7.el8.noarch.rpm
317 kB/s | 9.4 kB     00:00
(11/86): perl-Test-Harness-3.42-1.el8.noarch.rpm
4.5 MB/s | 279 kB     00:00
(12/86): perl-ExtUtils-Command-7.34-1.el8.noarch.rpm
520 kB/s |  19 kB     00:00

...

15 MB/s | 1.5 MB     00:00
--------------------------------------------------------------------------------
--------------------------------------------------------------------------------
----------------------------------------
Total
35 MB/s |  72 MB     00:02
Running transaction check
Transaction check succeeded.
Running transaction test
```

```
Transaction test succeeded.
Running transaction
  Preparing        :
1/1
  Running scriptlet: openssl-libs-1:1.1.1k-7.el8_6.x86_64
1/1
  Upgrading        : openssl-libs-1:1.1.1k-7.el8_6.x86_64
1/103
  Running scriptlet: openssl-libs-1:1.1.1k-7.el8_6.x86_64
1/103
  Upgrading        : libgcc-8.5.0-10.1.el8_6.x86_64
2/103
  Running scriptlet: libgcc-8.5.0-10.1.el8_6.x86_64
2/103
  Upgrading        : elfutils-libelf-0.186-1.el8.x86_64
3/103
  Installing       : perl-version-6:0.99.24-1.el8.x86_64
4/103
  Installing       : perl-CPAN-Meta-Requirements-2.140-396.el8.noarch
5/103
  Upgrading        : libsemanage-2.9-8.el8.x86_64
6/103
  Installing       : zlib-devel-1.2.11-17.el8.x86_64
7/103
  Installing       : python-srpm-macros-3-41.el8.noarch
8/103
  Installing       : python-rpm-macros-3-41.el8.noarch
9/103
  Installing       : python3-rpm-macros-3-41.el8.noarch
10/103
  Installing       : perl-Time-HiRes-4:1.9758-2.el8.x86_64
11/103
  Installing       : perl-ExtUtils-ParseXS-1:3.35-2.el8.noarch
12/103
  Installing       : perl-Test-Harness-1:3.42-1.el8.noarch
13/103
  Upgrading        : python3-libsemanage-2.9-8.el8.x86_64
14/103
  Upgrading        : policycoreutils-2.9-19.el8.x86_64
15/103
  Running scriptlet: policycoreutils-2.9-19.el8.x86_64
15/103
  Upgrading        : python3-policycoreutils-2.9-19.el8.noarch
16/103
  Installing       : dwz-0.12-10.el8.x86_64
17/103
```

```
  Installing       : ncurses-compat-libs-6.1-9.20180224.el8.x86_64
18/103
  Installing       : libesmtp-1.0.6-18.el8.x86_64
19/103
  Installing       : mailx-12.5-29.el8.x86_64
20/103
  Installing       : libkadm5-1.18.2-14.el8.x86_64
21/103
  Upgrading        : libgomp-8.5.0-10.1.el8_6.x86_64
22/103
  Running scriptlet: libgomp-8.5.0-10.1.el8_6.x86_64
22/103
  Upgrading        : platform-python-pip-9.0.3-22.el8.noarch
23/103
  Upgrading        : python3-pip-9.0.3-22.el8.noarch
24/103
  Upgrading        : python36-3.6.8-
38.module+el8.5.0+12207+5c5719bc.x86_64
25/103
  Running scriptlet: python36-3.6.8-
38.module+el8.5.0+12207+5c5719bc.x86_64
25/103
  Upgrading        : cpp-8.5.0-10.1.el8_6.x86_64
26/103
  Running scriptlet: cpp-8.5.0-10.1.el8_6.x86_64
26/103
  Upgrading        : gcc-8.5.0-10.1.el8_6.x86_64
27/103
  Running scriptlet: gcc-8.5.0-10.1.el8_6.x86_64
27/103
  Installing       : annobin-10.29-3.el8.x86_64
28/103
  Installing       : unzip-6.0-46.el8.x86_64
29/103
  Installing       : zip-3.0-23.el8.x86_64
30/103
  Installing       : perl-Math-Complex-1.59-421.el8.noarch
31/103
  Installing       : perl-Math-BigInt-1:1.9998.11-7.el8.noarch
32/103
  Installing       : perl-JSON-PP-1:2.97.001-3.el8.noarch
33/103
  Installing       : make-1:4.2.1-11.el8.x86_64
34/103
  Running scriptlet: make-1:4.2.1-11.el8.x86_64
34/103
```

```
  Installing       : libcom_err-devel-1.45.6-2.el8.x86_64
35/103
  Installing       : util-linux-user-2.32.1-28.el8.x86_64
36/103
  Installing       : libsepol-devel-2.9-3.el8.x86_64
37/103
  Installing       : pcre2-utf32-10.32-2.el8.x86_64
38/103
  Installing       : pcre2-utf16-10.32-2.el8.x86_64
39/103
  Installing       : pcre2-devel-10.32-2.el8.x86_64
40/103
  Installing       : libselinux-devel-2.9-5.el8.x86_64
41/103
  Installing       : patch-2.7.6-11.el8.x86_64
42/103
  Installing       : python3-pyparsing-2.1.10-7.el8.noarch
43/103
  Installing       : systemtap-sdt-devel-4.6-4.el8.x86_64
44/103
  Installing       : spax-1.5.3-13.el8.x86_64
45/103
  Running scriptlet: spax-1.5.3-13.el8.x86_64
45/103
  Installing       : m4-1.4.18-7.el8.x86_64
46/103
  Running scriptlet: m4-1.4.18-7.el8.x86_64
46/103
  Installing       : libverto-devel-0.3.0-5.el8.x86_64
47/103
  Installing       : bc-1.07.1-5.el8.x86_64
48/103
  Running scriptlet: bc-1.07.1-5.el8.x86_64
48/103
  Installing       : at-3.1.20-11.el8.x86_64
49/103
  Running scriptlet: at-3.1.20-11.el8.x86_64
49/103
  Installing       : keyutils-libs-devel-1.5.10-6.el8.x86_64
50/103
  Installing       : krb5-devel-1.18.2-14.el8.x86_64
51/103
  Installing       : time-1.9-3.el8.x86_64
52/103
  Running scriptlet: time-1.9-3.el8.x86_64
52/103
```

```
  Upgrading        : policycoreutils-python-utils-2.9-19.el8.noarch
80/103
  Installing       : elfutils-libelf-devel-0.186-1.el8.x86_64
81/103
  Upgrading        : elfutils-libs-0.186-1.el8.x86_64
82/103
  Upgrading        : mokutil-1:0.3.0-11.el8_6.1.x86_64
83/103
  Upgrading        : openssl-1:1.1.1k-7.el8_6.x86_64
84/103
  Installing       : kernel-devel-4.18.0-348.el8.x86_64
85/103
  Running scriptlet: kernel-devel-4.18.0-348.el8.x86_64

  ...

85/103
  Installing       : bzip2-1.0.6-26.el8.x86_64
86/103
  Cleanup          : policycoreutils-python-utils-2.9-14.el8.noarch
87/103
  Cleanup          : python3-policycoreutils-2.9-14.el8.noarch
88/103
  Cleanup          : python36-3.6.8-
2.module+el8.1.0+3334+5cb623d7.x86_64
89/103
  Running scriptlet: python36-3.6.8-
2.module+el8.1.0+3334+5cb623d7.x86_64
89/103
  Cleanup          : elfutils-libs-0.185-1.el8.x86_64
90/103
  Cleanup          : openssl-1:1.1.1k-4.el8.x86_64
91/103
  Cleanup          : python3-libsemanage-2.9-6.el8.x86_64
92/103
  Running scriptlet: gcc-8.4.1-1.el8.x86_64
93/103
  Cleanup          : gcc-8.4.1-1.el8.x86_64
93/103
  Running scriptlet: policycoreutils-2.9-14.el8.x86_64
94/103
  Cleanup          : policycoreutils-2.9-14.el8.x86_64
94/103
  Cleanup          : mokutil-1:0.3.0-11.el8.x86_64
95/103
```

```
  Cleanup          : python3-pip-9.0.3-19.el8.noarch
96/103
  Cleanup          : platform-python-pip-9.0.3-19.el8.noarch
97/103
  Cleanup          : openssl-libs-1:1.1.1k-4.el8.x86_64
98/103
  Running scriptlet: openssl-libs-1:1.1.1k-4.el8.x86_64
98/103
  Cleanup          : libsemanage-2.9-6.el8.x86_64
99/103
  Running scriptlet: cpp-8.4.1-1.el8.x86_64
100/103
  Cleanup          : cpp-8.4.1-1.el8.x86_64
100/103
  Cleanup          : libgcc-8.5.0-3.el8.x86_64
101/103
  Running scriptlet: libgcc-8.5.0-3.el8.x86_64
101/103
  Running scriptlet: libgomp-8.4.1-1.el8.x86_64
102/103
  Cleanup          : libgomp-8.4.1-1.el8.x86_64
102/103
  Running scriptlet: libgomp-8.4.1-1.el8.x86_64
102/103
  Cleanup          : elfutils-libelf-0.185-1.el8.x86_64
103/103
  Running scriptlet: elfutils-libelf-0.185-1.el8.x86_64
103/103
  Verifying        : esmtp-1.2-15.el8.x86_64
1/103
  Verifying        : libesmtp-1.0.6-18.el8.x86_64

  ...

Upgraded:
  cpp-8.5.0-10.1.el8_6.x86_64                              elfutils-
libelf-0.186-1.el8.x86_64     elfutils-libs-0.186-1.el8.x86_64
gcc-8.5.0-10.1.el8_6.x86_64
  libgcc-8.5.0-10.1.el8_6.x86_64                           libgomp-
8.5.0-10.1.el8_6.x86_64        libsemanage-2.9-8.el8.x86_64
mokutil-1:0.3.0-11.el8_6.1.x86_64
  openssl-1:1.1.1k-7.el8_6.x86_64                          openssl-
libs-1:1.1.1k-7.el8_6.x86_64   platform-python-pip-9.0.3-22.el8.noarch
policycoreutils-2.9-19.el8.x86_64
  policycoreutils-python-utils-2.9-19.el8.noarch           python3-
libsemanage-2.9-8.el8.x86_64   python3-pip-9.0.3-22.el8.noarch
```

```
python3-policycoreutils-2.9-19.el8.noarch
  python36-3.6.8-38.module+el8.5.0+12207+5c5719bc.x86_64
Installed:
  annobin-10.29-3.el8.x86_64                                    at-
3.1.20-11.el8.x86_64                          bc-1.07.1-5.el8.x86_64
  bzip2-1.0.6-26.el8.x86_64
cups-client-1:2.2.6-38.el8.x86_64                dwz-0.12-
10.el8.x86_64
  ed-1.14.2-4.el8.x86_64
efi-srpm-macros-3-3.el8.noarch                   elfutils-libelf-
devel-0.186-1.el8.x86_64
  esmtp-1.2-15.el8.x86_64
ghc-srpm-macros-1.4.2-7.el8.noarch               go-srpm-macros-2-
17.el8.noarch
  kernel-devel-4.18.0-348.el8.x86_64
keyutils-libs-devel-1.5.10-6.el8.x86_64          krb5-devel-1.18.2-
14.el8.x86_64
  libcom_err-devel-1.45.6-2.el8.x86_64
libesmtp-1.0.6-18.el8.x86_64                     libkadm5-1.18.2-
14.el8.x86_64
  liblockfile-1.14-1.el8.x86_64
libselinux-devel-2.9-5.el8.x86_64                libsepol-devel-2.9-
3.el8.x86_64
  libverto-devel-0.3.0-5.el8.x86_64                             m4-
1.4.18-7.el8.x86_64                           mailx-12.5-
29.el8.x86_64
  make-1:4.2.1-11.el8.x86_64
ncurses-compat-libs-6.1-9.20180224.el8.x86_64    ocaml-srpm-macros-
5-4.el8.noarch
  openblas-srpm-macros-2-2.el8.noarch
openssl-devel-1:1.1.1k-7.el8_6.x86_64            patch-2.7.6-
11.el8.x86_64
  pcre2-devel-10.32-2.el8.x86_64
pcre2-utf16-10.32-2.el8.x86_64                   pcre2-utf32-10.32-
2.el8.x86_64
  perl-CPAN-Meta-2.150010-396.el8.noarch
perl-CPAN-Meta-Requirements-2.140-396.el8.noarch  perl-CPAN-Meta-
YAML-0.018-397.el8.noarch
  perl-Encode-Locale-1.05-10.module+el8.3.0+6498+9eecfe51.noarch
perl-ExtUtils-Command-1:7.34-1.el8.noarch        perl-ExtUtils-
Install-2.14-4.el8.noarch
  perl-ExtUtils-MakeMaker-1:7.34-1.el8.noarch
perl-ExtUtils-Manifest-1.70-395.el8.noarch       perl-ExtUtils-
ParseXS-1:3.35-2.el8.noarch
  perl-JSON-PP-1:2.97.001-3.el8.noarch
perl-Math-BigInt-1:1.9998.11-7.el8.noarch        perl-Math-Complex-
```

```
1.59-421.el8.noarch
  perl-Test-Harness-1:3.42-1.el8.noarch
perl-Time-HiRes-4:1.9758-2.el8.x86_64              perl-devel-
4:5.26.3-419.el8_4.1.x86_64
  perl-srpm-macros-1-25.el8.noarch
perl-version-6:0.99.24-1.el8.x86_64                platform-python-
devel-3.6.8-41.el8.x86_64
  python-rpm-macros-3-41.el8.noarch
python-srpm-macros-3-41.el8.noarch                 python3-pyparsing-
2.1.10-7.el8.noarch
  python3-rpm-generators-5-7.el8.noarch
python3-rpm-macros-3-41.el8.noarch                 python36-devel-
3.6.8-38.module+el8.5.0+12207+5c5719bc.x86_64
  qt5-srpm-macros-5.15.2-1.el8.noarch
redhat-lsb-core-4.1-47.el8.x86_64                  redhat-lsb-submod-
security-4.1-47.el8.x86_64
  redhat-rpm-config-125-1.el8.noarch
rust-srpm-macros-5-2.el8.noarch                    spax-1.5.3-
13.el8.x86_64
  systemtap-sdt-devel-4.6-4.el8.x86_64
time-1.9-3.el8.x86_64                              unzip-6.0-
46.el8.x86_64
  util-linux-user-2.32.1-28.el8.x86_64
zip-3.0-23.el8.x86_64                              zlib-devel-1.2.11-
17.el8.x86_64

Complete!
OS package installations finished
+ Installing ONTAP Mediator. (Log: /tmp/ontap_mediator.JixKGP/ontap-
mediator-1.6.0/ontap-mediator-1.6.0/install_20221021155929.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_20221021155929.log)
+ WARNING: This system supports UEFI
           Secure Boot (SB) is currently disabled on this system.
           If SB is enabled in the future, SCST will not work unless
the following action is 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.modu
le-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 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@scs000099753 ~]# cat /etc/redhat-release
Red Hat Enterprise Linux release 8.5 (Ootpa)
[root@scs000099753 ~]#
```

Verify the installation

After the ONTAP Mediator has been installed, you should verify that the ONTAP Mediator services are running.

Steps

1. View the status of the ONTAP Mediator services:
   a. `systemctl status ontap_mediator`

```
[root@scspr1915530002 ~]# 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@scspr1915530002 ~]#
```

b. `systemctl status mediator-scst`

```
[root@scspr1915530002 ~]# 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@scspr1915530002 ~]#
```

2. Confirm the ports that are used by the ONTAP Mediator service:

`netstat`

```
[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
```

Post-installation configuration

After the ONTAP Mediator service is installed and running, additional configuration tasks must be performed in the ONTAP storage system to use the Mediator features:

- To use the ONTAP Mediator service in a MetroCluster IP configuration, see Configuring the ONTAP Mediator service from a MetroCluster IP configuration.
- To use SnapMirror active sync, see Install ONTAP Mediator Service and confirm the ONTAP cluster configuration.

Configure ONTAP Mediator security policies

The ONTAP Mediator server supports several configurable security settings. The default values for all settings are provide in a low_space_threshold_mib: 10read-only file:

`/opt/netapp/lib/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 the ONTAP Mediator service:

```
systemctl restart ontap_mediator
```

Modify ONTAP Mediator attributes

The following attributes can be configured:

Other default values in the `ontap_mediator.config.yaml` should not be modified.

- **Settings used to install third-party SSL certificates as replacements for the default self-signed certificates**

```
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'
cert_valid_days: '1095'                  # Used to set the expiration
on client certs to 3 years
x509_passin_pwd: 'pass:ontap'            # passphrase for the signed
client cert
```

- **Settings that provide protections 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
```

- **Fields that 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)
```

- **Setting that controls the required free space on the `/opt/netapp/lib/ontap_mediator` disk.**

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

```
low_space_threshold_mib: 10
```

- **Setting that controls RESERVE_LOG_SPACE.**

  The ONTAP Mediator server by default installation creates a separate disk space for the logs. The installer creates a new fixed-size file with a total of 700 MB of disk space to be used explicitly for 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
     ```

  2. Restart the Mediator:

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

```
RESERVE_LOG_SPACE=0
```

b. `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.

Manage the ONTAP mediator service

Manage the ONTAP Mediator service, including changing user credentials, stopping and re-enabling the service, verifying its health, and installing or uninstalling SCST for host maintenance. You can also manage certificates, such as regenerating self-signed certificates, replacing them with trusted third-party certificates, and troubleshooting certificate-related issues.

Change the username

You can change the username using the following procedure.

About this task

Perform this task on the Linux host on which the ONTAP Mediator service is installed.

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 on which the ONTAP Mediator service is installed.

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 ~]#
```

Stop the ONTAP Mediator service

To stop the ONTAP Mediator service, perform the following steps:

Steps

1. Stop the ONTAP Mediator:

   ```
   systemctl stop ontap_mediator
   ```

2. Stop SCST:

   ```
   systemctl stop mediator-scst
   ```

3. Disable the ONTAP Mediator and SCST:

   ```
   systemctl diable ontap_mediator mediator-scst
   ```

Re-enable the ONTAP Mediator service

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

Steps

1. Enable the 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 the ONTAP Mediator is healthy

After the ONTAP Mediator has been installed, you should verify that the ONTAP Mediator services are running.

Steps

1. View the status of the ONTAP Mediator services:

   a. `systemctl status ontap_mediator`

```
[root@scspr1915530002 ~]# 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@scspr1915530002 ~]#
```

b. `systemctl status mediator-scst`

```
[root@scspr1915530002 ~]# 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@scspr1915530002 ~]#
```

2. Confirm the ports that are used by the ONTAP Mediator service:

`netstat`

```
[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
```

Manually uninstall SCST to perform host maintenance

To uninstall SCST, you need the SCST tar bundle that is used for the installed version of ONTAP Mediator.

Steps

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

For this version …	Use this tar bundle…
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

2. Issue the following 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`

Manually install SCST to perform host maintenance

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

1. Issue the following 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`
   g. `cp scst/src/certs/scst_module_key.der /opt/netapp/lib/ontap_mediator/ontap_mediator/SCST_mod_keys/.`
   h. `cp scst/src/certs/scst_module_key.der /opt/netapp/lib/ontap_mediator/ontap_mediator/SCST_mod_keys/.`
   i. `patch /etc/init.d/scst < /opt/netapp/lib/ontap_mediator/systemd/scst.patch`
2. Optionally, if Secure Boot is enabled, before you reboot, perform the following steps:
   a. Determine each file name for "scst_vdisk", "scst", and "iscsi_scst" modules:

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

   b. Determine the kernel release:

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

   c. Sign each 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_modu
le_key.priv \
/opt/netapp/lib/ontap_mediator/ontap_mediator/SCST_mod_keys/scst_modu
le_key.der \
_module-filename_
```

   d. Install the correct key with the UEFI 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
```

3. Perform a reboot:

```
reboot
```

Uninstall the ONTAP Mediator service

If necessary, you can remove the ONTAP Mediator service.

Before you begin

The ONTAP Mediator must be disconnected from ONTAP before you remove the ONTAP Mediator service.

About this task

You need to perform this task on the Linux host on which the ONTAP Mediator service is installed.

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 the ONTAP Mediator service:

```
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 ~]#
```

Regenerate a temporary self-signed certificate

You can regenerate a temporary self-signed certificate using the following procedure.

About this task

- You perform this task on the Linux host on which the ONTAP Mediator service is installed.
- 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 the 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 the ONTAP Mediator service:

   ```
   ./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 self-signed certificates with trusted third-party certificates

You can replace self-signed certificates with trusted third-party certificates.

About this task

- You perform this task on the Linux host on which the ONTAP Mediator service is installed.
- 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.

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, namely `ca.key`, `ca.csr`, `ca.srl`, and `ca.crt` located at `/opt/netapp/lib/ontap_mediator/ontap_mediator/server_config/` with the third-party certificate actors.

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

Steps

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@scs000216655 server_config]# openssl req -key ca.key -new -config
   openssl_ca.cnf -out ca.csr
   Enter pass phrase for ca.key:
   [root@scs000216655 server_config]# cat ca.csr
   -----BEGIN CERTIFICATE REQUEST-----
   MIIE6TCCAtECAQAwgaMxCzAJBgNVBAYTAlVTMRMwEQYDVQQIDApDYWxpZm9ybmlh
   ...
   erARKhY9z0e8BHPl3g==
   -----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`.

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

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

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.

Step

To generate a server certificate, run the following commands from the folder `/opt/netapp/lib/ontap_mediator/ontap_mediator/server_config`:

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

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
```

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

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

The certificate configuration is supplied to the ONTAP Mediator service 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 the `ca.*` files with the third-party certificates.
2. Restart the 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 `ca.*` or store the third-party certificates in a different location.

Steps

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

   For example, if you obtain `intermediate.crt` from the PKI authority and store its private key `intermediate.key` and certificate signing request `intermediate.csr` at a location `/opt/netapp/lib/ontap_mediator/ontap_mediator/server_config`, then the user_config file should look like the following:

```
[root@scs000216655 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'
```

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

```
systemctl restart ontap_mediator
```

Troubleshoot certificate-related issues

You can check certain properties of the certificates.

Verify certificate expiration

Use the following command to identify the certificate validity range:

```
[root@scs000216982 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 `ca.crt`.

```
[root@scs000216982 server_config]# pwd
/opt/netapp/lib/ontap_mediator/ontap_mediator/server_config

[root@scs000216982 server_config]# cat openssl_ca.cnf
...
[ v3_ca ]
subjectKeyIdentifier = hash
authorityKeyIdentifier = keyid:always,issuer
basicConstraints = critical, CA:true
keyUsage = critical, cRLSign, digitalSignature, keyCertSign

[root@scs000216982 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 the ONTAP Mediator is installed.

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

```
[root@scs000216982 server_config]# pwd
/opt/netapp/lib/ontap_mediator/ontap_mediator/server_config

[root@scs000216982 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@scs000216982 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@scs000216982 server_config]# openssl rsa -noout -modulus -in
intermediate.key | openssl md5
Enter pass phrase for intermediate.key:
(stdin)= 14c6b98b0c7c59012b1de89eee4a9dbc
[root@scs000216982 server_config]# openssl x509 -noout -modulus -in
intermediate.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@scs000216982 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 OS host for ONTAP Mediator

For optimal performance, you should maintain the host OS for ONTAP Mediator on a regular basis.

Reboot the host

Reboot the host when the clusters are healthy. While the ONTAP Mediator is offline, the clusters are at risk of not being able to react properly to failures. A service window is recommended if a reboot is required.

ONTAP Mediator will automatically resume during a reboot and will re-enter the relationships that were previously configured with ONTAP clusters.

Host package updates

Any library or yum packages (except the kernel) can be safely updated but might require a reboot to take effect. A service window is recommended if a reboot is required.

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

You should reboot if any of the ONTAP Mediator dependencies are updated because they will not take immediate effect on running processes.

Host OS minor kernel upgrades

SCST must be compiled for the kernel that is being used. To update the OS, a maintenance window is required.

Steps

Perform the following steps to upgrade the host OS kernel.

1. Stop the ONTAP Mediator
2. Uninstall the SCST package. (SCST doesn’t provide an upgrade mechanism.)
3. Upgrade the OS, and reboot.
4. Re-install the SCST package.
5. Re-enable the ONTAP Mediator services.

Host changes to the hostname or IP

About this task

- You perform this task on the Linux host on which the ONTAP Mediator service is installed.
- 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 the 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 the 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
```

Copyright information

Trademark information