# Table of Contents

Install or upgrade ........................................................................................................................................... 1
  Prepare to install or upgrade the ONTAP Mediator service ........................................................................ 1
  Upgrade the host operating system and then the ONTAP Mediator ......................................................... 2
  Enable access to the repositories ............................................................................................................. 9
  Download the Mediator installation package ............................................................................................ 14
  Verify the ONTAP Mediator code signature ............................................................................................. 15
  Install the ONTAP Mediator installation package .................................................................................. 20
  Verify the installation ............................................................................................................................... 37
  Post-installation configuration ............................................................................................................... 38
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 SM-BC relationship.

Before you begin

You must meet the following prerequisites.

<table>
<thead>
<tr>
<th>ONTAP Mediator version</th>
<th>Supported Linux versions</th>
</tr>
</thead>
</table>
| 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:

<table>
<thead>
<tr>
<th>All RHEL/CentOS versions</th>
<th>Additional packages for RHEL 8.x / Rocky Linux 8</th>
<th>Additional packages for RHEL 9.x / Rocky Linux 9</th>
</tr>
</thead>
<tbody>
<tr>
<td>• openssl</td>
<td>• python3-pip</td>
<td>• python3-pip</td>
</tr>
<tr>
<td>• openssl-devel</td>
<td>• elfutils-libelf-devel</td>
<td>• elfutils-libelf-devel</td>
</tr>
<tr>
<td>• kernel-devel-$ (uname -r)</td>
<td>• policycoreutils-python-utils</td>
<td>• policycoreutils-python-utils</td>
</tr>
<tr>
<td>• gcc</td>
<td>• redhat-lsb-core</td>
<td>• python3</td>
</tr>
<tr>
<td>• make</td>
<td>• python39</td>
<td>• python3-devel</td>
</tr>
<tr>
<td>• libselinux-utils</td>
<td>• python39-devel</td>
<td></td>
</tr>
<tr>
<td>• patch</td>
<td></td>
<td></td>
</tr>
<tr>
<td>• bzip2</td>
<td></td>
<td></td>
</tr>
<tr>
<td>• perl-Data-Dumper</td>
<td></td>
<td></td>
</tr>
<tr>
<td>• perl-ExtUtils-MakeMake</td>
<td></td>
<td></td>
</tr>
<tr>
<td>• efiloarmgr</td>
<td></td>
<td></td>
</tr>
<tr>
<td>• mokutil</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

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.
3. Reinstall ONTAP Mediator.

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

```bash
[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.

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

5. Replace the log files.

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

6. Start the ontap_mediator service.

```bash
[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...
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
```
**Procedure for SnapMirror Business Continuity**

For SnapMirror Business Continuity, 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

<table>
<thead>
<tr>
<th>Mediator Address</th>
<th>Peer Cluster</th>
<th>Connection Status</th>
<th>Quorum Status</th>
</tr>
</thead>
<tbody>
<tr>
<td>172.31.49.237</td>
<td>peer2</td>
<td>unreachable</td>
<td>true</td>
</tr>
</tbody>
</table>

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

<table>
<thead>
<tr>
<th>Job ID</th>
<th>Name</th>
<th>Vserver</th>
<th>Node</th>
<th>State</th>
</tr>
</thead>
<tbody>
<tr>
<td>39</td>
<td>mediator remove</td>
<td>peer1</td>
<td>peer1-node1</td>
<td>Success</td>
</tr>
</tbody>
</table>

Description: Removing entry in mediator

peer1::> security certificate show -common-name ONTAPMediatorCA

<table>
<thead>
<tr>
<th>Vserver</th>
<th>Serial Number</th>
<th>Certificate Name Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>peer1</td>
<td>4A790360081F41145E14C5D7CE721DC6C210007F</td>
<td>ONTAPMediatorCA server-ca</td>
</tr>
</tbody>
</table>

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

<table>
<thead>
<tr>
<th>Job ID</th>
<th>Name</th>
<th>Owning Vserver</th>
<th>Node</th>
<th>State</th>
</tr>
</thead>
<tbody>
<tr>
<td>43</td>
<td>mediator add</td>
<td>peer1</td>
<td>peer1-node2</td>
<td>Success</td>
</tr>
</tbody>
</table>

Description: Creating a mediator entry

peer1::> snapmirror mediator show

Mediator Address Peer Cluster    Connection Status Quorum Status
--------------------------------------------------------------------
172.31.49.237                  peer2               connected       true
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:

<table>
<thead>
<tr>
<th>If your operating system is...</th>
<th>You must provide access to these repositories...</th>
</tr>
</thead>
<tbody>
<tr>
<td>RHEL 7.x</td>
<td>• rhel-7-server-optional-rpms</td>
</tr>
<tr>
<td>RHEL 8.x</td>
<td>• rhel-8-for-x86_64-baseos-rpms</td>
</tr>
<tr>
<td></td>
<td>• rhel-8-for-x86_64-appstream-rpms</td>
</tr>
<tr>
<td>RHEL 9.x</td>
<td>• rhel-9-for-x86_64-baseos-rpms</td>
</tr>
<tr>
<td></td>
<td>• rhel-9-for-x86_64-appstream-rpms</td>
</tr>
<tr>
<td>CentOS 7.x</td>
<td>• C7.6.1810 - Base repository</td>
</tr>
<tr>
<td>Rocky Linux 8</td>
<td>• appstream</td>
</tr>
<tr>
<td></td>
<td>• baseos</td>
</tr>
<tr>
<td>Rocky Linux 9</td>
<td>• appstream</td>
</tr>
<tr>
<td></td>
<td>• baseos</td>
</tr>
</tbody>
</table>

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:
   
   ```bash
   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:

   ```bash
   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:

   ```bash
   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:

<table>
<thead>
<tr>
<th>File</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ONTAP-Mediator-development.pub</td>
<td>The public key used to verify the signature</td>
</tr>
<tr>
<td>csc-prod-chain-ONTAP-Mediator.pem</td>
<td>The public certification CA chain of trust</td>
</tr>
<tr>
<td>csc-prod-ONTAP-Mediator.pem</td>
<td>The certificate used to generate the key</td>
</tr>
<tr>
<td>ontap-mediator-1.7.0</td>
<td>The product installation executable for version 1.7.0</td>
</tr>
<tr>
<td>ontap-mediator-1.7.0.sig</td>
<td>The SHA-256 hashed, then RSA-signed using the csc-prod key, signature for the installer</td>
</tr>
<tr>
<td>ontap-mediator-1.7.0.sig.tsr</td>
<td>The revocation request for use by OCSCP for the installer’s signature</td>
</tr>
<tr>
<td>tsa-prod-ONTAP-Mediator.pem</td>
<td>The public certificate for the TSR</td>
</tr>
<tr>
<td>tsa-prod-chain-ONTAP-Mediator.pem</td>
<td>The public certificate CA Chain for the TSR</td>
</tr>
</tbody>
</table>

Steps

   a. Find the OCSP URL used to register the certificate because developer certificates might not provide a uri.

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

   b. Generate an OCSP request for the certificate.

   ```bash
   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:

   ```bash
   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)

```bash
[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/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
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
  e7:6c:8c:49:dd:0c:fd:8e:20:08:0d:0f:5a:29:a3:19:03:
  97:3a:0b:0a:8e:a3:9e:e3:f4:e0:d6:1a:c9:b5:14:8c:3e:54:
  1f:86:a9:16:ce:dd:9a:8b:3a:ff:3f:4c:48:8d:ad:de:e8:3e:3c:
  cl: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
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
```

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:
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.

<table>
<thead>
<tr>
<th>Package</th>
<th>Architecture</th>
<th>Repository</th>
</tr>
</thead>
<tbody>
<tr>
<td>bzip2</td>
<td>x86_64</td>
<td>rhel-8-for-</td>
</tr>
<tr>
<td>1.0.6-26.el8</td>
<td></td>
<td></td>
</tr>
<tr>
<td>x86_64-baseos-rpms</td>
<td>60 k</td>
<td>rhel-8-for-</td>
</tr>
<tr>
<td>elfutils-libelf-devel</td>
<td>x86_64</td>
<td></td>
</tr>
<tr>
<td>0.186-1.el8</td>
<td></td>
<td></td>
</tr>
<tr>
<td>x86_64-baseos-rpms</td>
<td>60 k</td>
<td></td>
</tr>
<tr>
<td>kernel-devel</td>
<td>x86_64</td>
<td></td>
</tr>
<tr>
<td>4.18.0-348.el8</td>
<td></td>
<td></td>
</tr>
<tr>
<td>x86_64-baseos-rpms</td>
<td>20 M</td>
<td></td>
</tr>
<tr>
<td>make</td>
<td>x86_64</td>
<td></td>
</tr>
<tr>
<td>1:4.2.1-11.el8</td>
<td></td>
<td></td>
</tr>
<tr>
<td>x86_64-baseos-rpms</td>
<td>498 k</td>
<td></td>
</tr>
<tr>
<td>openssl-devel</td>
<td>x86_64</td>
<td></td>
</tr>
<tr>
<td>1:1.1.1k-7.el8_6</td>
<td></td>
<td></td>
</tr>
<tr>
<td>x86_64-baseos-rpms</td>
<td>2.3 M</td>
<td></td>
</tr>
<tr>
<td>patch</td>
<td>x86_64</td>
<td></td>
</tr>
<tr>
<td>2.7.6-11.el8</td>
<td></td>
<td></td>
</tr>
<tr>
<td>x86_64-baseos-rpms</td>
<td>138 k</td>
<td></td>
</tr>
<tr>
<td>perl-ExtUtils-MakeMaker</td>
<td>noarch</td>
<td></td>
</tr>
<tr>
<td>1:7.34-1.el8</td>
<td></td>
<td></td>
</tr>
<tr>
<td>x86_64-appstream-rpms</td>
<td>301 k</td>
<td></td>
</tr>
<tr>
<td>python36-devel</td>
<td>x86_64</td>
<td></td>
</tr>
<tr>
<td>3.6.8-38.module+el8.5.0+12207+5c5719bc</td>
<td></td>
<td></td>
</tr>
<tr>
<td>x86_64-appstream-rpms</td>
<td>17 k</td>
<td></td>
</tr>
<tr>
<td>redhat-lsb-core</td>
<td>x86_64</td>
<td></td>
</tr>
<tr>
<td>4.1-47.el8</td>
<td></td>
<td></td>
</tr>
<tr>
<td>x86_64-appstream-rpms</td>
<td>45 k</td>
<td></td>
</tr>
<tr>
<td>Upgrading:</td>
<td></td>
<td></td>
</tr>
<tr>
<td>cpp</td>
<td>x86_64</td>
<td></td>
</tr>
<tr>
<td>8.5.0-10.1.el8_6</td>
<td></td>
<td></td>
</tr>
<tr>
<td>x86_64-appstream-rpms</td>
<td>10 M</td>
<td></td>
</tr>
<tr>
<td>elfutils-libelf</td>
<td>x86_64</td>
<td></td>
</tr>
</tbody>
</table>
0.186-1.el8
x86_64-baseos-rpms 229 k
elfutils-libs x86_64
0.186-1.el8
x86_64-baseos-rpms 295 k
gcc x86_64
8.5.0-10.1.el8_6
x86_64-appstream-rpms 23 M
libgcc x86_64
8.5.0-10.1.el8_6
x86_64-baseos-rpms 80 k
libgomp x86_64
8.5.0-10.1.el8_6
x86_64-baseos-rpms 207 k
libsemanage x86_64
2.9-8.el8
x86_64-baseos-rpms 168 k
mokutil x86_64
1:0.3.0-11.el8_6.1
x86_64-baseos-rpms 46 k
openssl x86_64
1:1.1.1k-7.el8_6
x86_64-baseos-rpms 709 k
openssl-lib x86_64
1:1.1.1k-7.el8_6
x86_64-baseos-rpms 1.5 M
platform-python-pip noarch
9.0.3-22.el8
x86_64-baseos-rpms 1.6 M
policycoreutils x86_64
2.9-19.el8
x86_64-baseos-rpms 374 k
policycoreutils-python-utils noarch
2.9-19.el8
x86_64-baseos-rpms 253 k
python3-libsemanage x86_64
2.9-8.el8
x86_64-baseos-rpms 128 k
python3-pip noarch
9.0.3-22.el8
x86_64-appstream-rpms 20 k
python3-policycoreutils noarch
2.9-19.el8
x86_64-baseos-rpms 2.2 M
python36 x86_64
3.6.8-38.module+el8.5.0+12207+5c5719bc rhel-8-for-
<table>
<thead>
<tr>
<th>Package</th>
<th>Size</th>
<th>Version</th>
<th>Source</th>
</tr>
</thead>
<tbody>
<tr>
<td>x86_64-appstream-rpms</td>
<td>19 k</td>
<td>19-3.el8</td>
<td>rhel-8-for-</td>
</tr>
<tr>
<td>annobin</td>
<td></td>
<td>10.29</td>
<td></td>
</tr>
<tr>
<td>at</td>
<td>117 k</td>
<td>3.1.20-11.el8</td>
<td></td>
</tr>
<tr>
<td>bc</td>
<td>81 k</td>
<td>1.07.1-5.el8</td>
<td></td>
</tr>
<tr>
<td>cups-client</td>
<td>129 k</td>
<td>1:2.2.6-38.el8</td>
<td></td>
</tr>
<tr>
<td>dwz</td>
<td>169 k</td>
<td>0.12-10.el8</td>
<td></td>
</tr>
<tr>
<td>ed</td>
<td>109 k</td>
<td>1.14.2-4.el8</td>
<td></td>
</tr>
<tr>
<td>efi-srpm-macros</td>
<td>82 k</td>
<td>3-3.el8</td>
<td></td>
</tr>
<tr>
<td>esmtp</td>
<td>22 k</td>
<td>1.2-15.el8</td>
<td>EPEL-8</td>
</tr>
<tr>
<td>ghc-srpm-macros</td>
<td>57 k</td>
<td>1.4.2-7.el8</td>
<td></td>
</tr>
<tr>
<td>go-srpm-macros</td>
<td>9.4 k</td>
<td>2-17.el8</td>
<td></td>
</tr>
<tr>
<td>keyutils-libs-devel</td>
<td>13 k</td>
<td>1.5.10-6.el8</td>
<td></td>
</tr>
<tr>
<td>krb5-devel</td>
<td>48 k</td>
<td>1.18.2-14.el8</td>
<td></td>
</tr>
<tr>
<td>libcom_err-devel</td>
<td>560 k</td>
<td>1.45.6-2.el8</td>
<td></td>
</tr>
<tr>
<td>libesmtp</td>
<td>38 k</td>
<td>1.18.64-baseos-rpms</td>
<td></td>
</tr>
<tr>
<td>libkadm5</td>
<td>70 k</td>
<td>1.0.6-18.el8</td>
<td>EPEL-8</td>
</tr>
<tr>
<td></td>
<td></td>
<td>1.18.2-14.el8</td>
<td></td>
</tr>
<tr>
<td>Package</td>
<td>Version</td>
<td>Size</td>
<td></td>
</tr>
<tr>
<td>-------------------------------</td>
<td>---------------</td>
<td>--------</td>
<td></td>
</tr>
<tr>
<td>libblockfile</td>
<td>1.14-1.el8</td>
<td>x86_64</td>
<td></td>
</tr>
<tr>
<td>x86_64-appstream-rpms</td>
<td>2.9-5.el8</td>
<td>rhel-8-for-x86_64</td>
<td></td>
</tr>
<tr>
<td>libselinux-devel</td>
<td>2.9-3.el8</td>
<td>rhel-8-for-x86_64</td>
<td></td>
</tr>
<tr>
<td>x86_64-baseos-rpms</td>
<td>0.3.0-5.el8</td>
<td>rhel-8-for-x86_64</td>
<td></td>
</tr>
<tr>
<td>m4</td>
<td>1.4.18-7.el8</td>
<td>x86_64</td>
<td></td>
</tr>
<tr>
<td>mailx</td>
<td>12.5-29.el8</td>
<td>x86_64</td>
<td></td>
</tr>
<tr>
<td>ncurses-compat-libs</td>
<td>6.1-9.20180224.el8</td>
<td>x86_64</td>
<td></td>
</tr>
<tr>
<td>ocaml-srpm-macros</td>
<td>5-4.el8</td>
<td>noarch</td>
<td></td>
</tr>
<tr>
<td>openblas-srpm-macros</td>
<td>2-2.el8</td>
<td>noarch</td>
<td></td>
</tr>
<tr>
<td>pcire2-devel</td>
<td>10.32-2.el8</td>
<td>x86_64</td>
<td></td>
</tr>
<tr>
<td>pcre2-utf16</td>
<td>10.32-2.el8</td>
<td>x86_64</td>
<td></td>
</tr>
<tr>
<td>pcre2-utf32</td>
<td>10.32-2.el8</td>
<td>x86_64</td>
<td></td>
</tr>
<tr>
<td>perl-CPAN-Meta-YAML</td>
<td>0.018-397.el8</td>
<td>noarch</td>
<td></td>
</tr>
<tr>
<td>perl-ExtUtils-Command</td>
<td>1:7.34-1.el8</td>
<td>noarch</td>
<td></td>
</tr>
<tr>
<td>perl-ExtUtils-Install</td>
<td>2.14-4.el8</td>
<td>noarch</td>
<td></td>
</tr>
<tr>
<td>perl-ExtUtils-Command</td>
<td>1:7.34-1.el8</td>
<td>noarch</td>
<td></td>
</tr>
<tr>
<td>perl-ExtUtils-Install</td>
<td>2.14-4.el8</td>
<td>noarch</td>
<td></td>
</tr>
</tbody>
</table>

* Size is in KB for each package.*
<table>
<thead>
<tr>
<th>Package</th>
<th>Version</th>
<th>Architecture</th>
<th>Repository</th>
<th>Size</th>
</tr>
</thead>
<tbody>
<tr>
<td>perl-ExtUtils-Manifest</td>
<td>noarch</td>
<td>1.70-395.el8</td>
<td>rhel-8-for-</td>
<td></td>
</tr>
<tr>
<td>x86_64-appstream-rpms</td>
<td>37 k</td>
<td>perl-ExtUtils-ParseXS</td>
<td>noarch</td>
<td></td>
</tr>
<tr>
<td>perl-JSON-PP</td>
<td>noarch</td>
<td>1:2.97.001-3.el8</td>
<td>rhel-8-for-</td>
<td></td>
</tr>
<tr>
<td>perl-Math-BigInt</td>
<td>noarch</td>
<td>1:1.9998.11-7.el8</td>
<td>rhel-8-for-</td>
<td></td>
</tr>
<tr>
<td>perl-Math-Complex</td>
<td>noarch</td>
<td>1:3.35-2.el8</td>
<td>rhel-8-for-</td>
<td></td>
</tr>
<tr>
<td>perl-Math-Complex</td>
<td>noarch</td>
<td>1:3.35-2.el8</td>
<td>rhel-8-for-</td>
<td></td>
</tr>
<tr>
<td>perl-Test-Harness</td>
<td>noarch</td>
<td>1:3.42-1.el8</td>
<td>rhel-8-for-</td>
<td></td>
</tr>
<tr>
<td>perl-devel</td>
<td>x86_64</td>
<td>4:5.26.3-419.el8</td>
<td>rhel-8-for-</td>
<td></td>
</tr>
<tr>
<td>perl-srpm-macros</td>
<td>noarch</td>
<td>1-25.el8</td>
<td>rhel-8-for-</td>
<td></td>
</tr>
<tr>
<td>perl-version</td>
<td>x86_64</td>
<td>6:0.99.24-1.el8</td>
<td>rhel-8-for-</td>
<td></td>
</tr>
<tr>
<td>platform-python-devel</td>
<td>x86_64</td>
<td>3.6.8-41.el8</td>
<td>rhel-8-for-</td>
<td></td>
</tr>
<tr>
<td>perl-srpm-macros</td>
<td>noarch</td>
<td>3-41.el8</td>
<td>rhel-8-for-</td>
<td></td>
</tr>
<tr>
<td>perl-srpm-macros</td>
<td>noarch</td>
<td>3-41.el8</td>
<td>rhel-8-for-</td>
<td></td>
</tr>
<tr>
<td>perl-srpm-macros</td>
<td>noarch</td>
<td>3-41.el8</td>
<td>rhel-8-for-</td>
<td></td>
</tr>
<tr>
<td>perl-srpm-macros</td>
<td>noarch</td>
<td>3-41.el8</td>
<td>rhel-8-for-</td>
<td></td>
</tr>
<tr>
<td>perl-srpm-macros</td>
<td>noarch</td>
<td>3-41.el8</td>
<td>rhel-8-for-</td>
<td></td>
</tr>
<tr>
<td>perl-srpm-macros</td>
<td>noarch</td>
<td>3-41.el8</td>
<td>rhel-8-for-</td>
<td></td>
</tr>
<tr>
<td>python-rpm-macros</td>
<td>noarch</td>
<td>3-41.el8</td>
<td>rhel-8-for-</td>
<td></td>
</tr>
<tr>
<td>python-srpm-macros</td>
<td>noarch</td>
<td>3-41.el8</td>
<td>rhel-8-for-</td>
<td></td>
</tr>
<tr>
<td>python3-pyparsing</td>
<td>noarch</td>
<td>2.1.10-7.el8</td>
<td>rhel-8-for-</td>
<td></td>
</tr>
<tr>
<td>python3-rpm-generators</td>
<td>noarch</td>
<td>5-7.el8</td>
<td>rhel-8-for-</td>
<td></td>
</tr>
<tr>
<td>python3-rpm-macros</td>
<td>noarch</td>
<td>3-41.el8</td>
<td>rhel-8-for-</td>
<td></td>
</tr>
<tr>
<td>qt5-srpm-macros</td>
<td>noarch</td>
<td>3-41.el8</td>
<td>rhel-8-for-</td>
<td></td>
</tr>
<tr>
<td>Package</td>
<td>Size</td>
<td>Architecture</td>
<td></td>
<td></td>
</tr>
<tr>
<td>----------------------------------------------</td>
<td>-------</td>
<td>----------------</td>
<td></td>
<td></td>
</tr>
<tr>
<td>5.15.2-1.el8</td>
<td>11 k</td>
<td>x86_64</td>
<td></td>
<td></td>
</tr>
<tr>
<td>x86_64-appstream-rpms</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>redhat-1sb-submod-security</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>4.1-47.el8</td>
<td>22 k</td>
<td>noarch</td>
<td></td>
<td></td>
</tr>
<tr>
<td>x86_64-appstream-rpms</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>redhat-rpm-config</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>125-1.el8</td>
<td>87 k</td>
<td>noarch</td>
<td></td>
<td></td>
</tr>
<tr>
<td>x86_64-appstream-rpms</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>rust-srpm-macros</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>5-2.el8</td>
<td>9.3 k</td>
<td>x86_64</td>
<td></td>
<td></td>
</tr>
<tr>
<td>x86_64-appstream-rpms</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>spax</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1.5.3-13.el8</td>
<td>217 k</td>
<td>x86_64</td>
<td></td>
<td></td>
</tr>
<tr>
<td>x86_64-baseos-rpms</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>systemtap-sdt-devel</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>4.6-4.el8</td>
<td>86 k</td>
<td>x86_64</td>
<td></td>
<td></td>
</tr>
<tr>
<td>x86_64-appstream-rpms</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>time</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1.9-3.el8</td>
<td>54 k</td>
<td>x86_64</td>
<td></td>
<td></td>
</tr>
<tr>
<td>x86_64-baseos-rpms</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>unzip</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>6.0-46.el8</td>
<td>196 k</td>
<td>x86_64</td>
<td></td>
<td></td>
</tr>
<tr>
<td>x86_64-baseos-rpms</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>util-linux-user</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2.32.1-28.el8</td>
<td>100 k</td>
<td>x86_64</td>
<td></td>
<td></td>
</tr>
<tr>
<td>x86_64-baseos-rpms</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>zip</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>3.0-23.el8</td>
<td>270 k</td>
<td>x86_64</td>
<td></td>
<td></td>
</tr>
<tr>
<td>x86_64-baseos-rpms</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>zlib-devel</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1.2.11-17.el8</td>
<td>58 k</td>
<td>x86_64</td>
<td></td>
<td></td>
</tr>
<tr>
<td>x86_64-baseos-rpms</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Installing weak dependencies:</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>perl-CPAN-Meta</td>
<td></td>
<td>noarch</td>
<td></td>
<td></td>
</tr>
<tr>
<td>2.150010-396.el8</td>
<td>191 k</td>
<td>x86_64</td>
<td></td>
<td></td>
</tr>
<tr>
<td>x86_64-appstream-rpms</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>perl-CPAN-Meta-Requirements</td>
<td></td>
<td>noarch</td>
<td></td>
<td></td>
</tr>
<tr>
<td>2.140-396.el8</td>
<td>37 k</td>
<td>noarch</td>
<td></td>
<td></td>
</tr>
<tr>
<td>x86_64-appstream-rpms</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>perl-Encode-Locale</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1.05-10.module+el8.3.0+6498+9eecfe51</td>
<td>22 k</td>
<td>x86_64</td>
<td></td>
<td></td>
</tr>
<tr>
<td>x86_64-appstream-rpms</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>perl-Time-HiRes</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>4:1.9758-2.el8</td>
<td>61 k</td>
<td>x86_64</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Transaction Summary
=======================================================================
=======================================================================
==========================================
Install  69 Packages
Upgrade  17 Packages

Total download size: 72 M
Is this ok [y/N]: y
Download 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
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
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
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:
  1/103

Running scriptlet: openssl-libs-1:1.1.1k-7.el8_6.x86_64
1/103

Upgrading:
  1/103

Upgrading:
  1/103

Upgrading:
  1/103

Running scriptlet: libgcc-8.5.0-10.1.el8_6.x86_64
2/103

Upgrading:
  2/103

Upgrading:
  2/103

Installing:
  3/103

Installing:
  4/103

Installing:
  5/103

Upgrading:
  6/103

Installing:
  7/103

Installing:
  8/103

Installing:
  9/103

Installing:
  10/103

Installing:
  11/103

Installing:
  12/103

Installing:
  13/103

Upgrading:
  14/103

Upgrading:
  15/103

Running scriptlet: policycoreutils-2.9-19.el8.x86_64
15/103

Upgrading:
  15/103

Upgrading:
  16/103

Installing:
  17/103
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: systemd-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.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: openssl1-1:1.1.1k-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-
  lib-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
Installed:
  annobin-10.29-3.el8.x86_64
  bc-1.07.1-5.el8.x86_64
  bzip2-1.0.6-26.el8.x86_64
dwz-0.12-
  cups-client-1:2.2.6-38.el8.x86_64
  ed-1.14.2-4.el8.x86_64
  elfutils-libelf-
  efibootloader-0.186-1.el8.x86_64
  efisrpm-macros-3-3.el8.noarch
  go-srpm-macros-2-
  kernel-devel-4.18.0-348.el8.x86_64
  krb5-devel-1.18.2-
  ld-2.33.1-32.el8.x86_64
  libcom_err-devel-1.45.6-2.el8.x86_64
  libkadm5-1.18.2-
  libkpie-devel-6.4-1.el8.x86_64
  libkpie-libs-devel-6.4-1.el8.x86_64
  libsepol-devel-2.9-
  libsepol-1.18.2-13.el8.x86_64
  libsepol-devel-2.9-
  libverto-devel-0.3.0-5.el8.x86_64
  mailx-12.5-
  make-4.2.1-11.el8.x86_64
  openssl-devel-1:1.1.1k-7.el8.x86_64
  patch-2.7.6-
  pcre2-utf16-10.32-2.el8.x86_64
  pcre2-utf32-10.32-2.el8.x86_64
  perl-CPAN-Meta-Perl-1.2-1.png
  perl-CPAN-Meta-YAML-1.15-4.png
  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-1:7.0-395.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-
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.module-signing to sign the SCST kernel module. Note that reboot will be
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
```

```
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
```

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 Business Continuity, 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:

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

  ```yaml
  cert_path: '/opt/netapp/lib/ontap_mediator/ontap_mediator/server_config/ontap_mediator_server.crt'
  key_path: '/opt/netapp/lib/ontap_mediator/ontap_mediator/server_config/ontap_mediator_server.key'
  ca_cert_path: '/opt/netapp/lib/ontap_mediator/ontap_mediator/server_config/ca.crt'
  ca_key_path: '/opt/netapp/lib/ontap_mediator/ontap_mediator/server_config/ca.key'
  ca_serial_path: '/opt/netapp/lib/ontap_mediator/ontap_mediator/server_config/ca.srl'
  cert_valid_days: '1095'  # 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:
    ```yaml
    authentication_lock_window_seconds: 300
    ```
  - Lock the account if five failures occur within the window timeframe:
    ```yaml
    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.
**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:
   ```bash
   /opt/netapp/lib/ontap_mediator/tools/mediator_env
   ```

2. Restart the Mediator:
   ```bash
   a. cat /opt/netapp/lib/ontap_mediator/tools/mediator_env | grep "RESERVE_LOG_SPACE"
   ```
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.