



ONTAP Mediator 文档

ONTAP Mediator

NetApp
October 19, 2026

This PDF was generated from <https://docs.netapp.com/zh-cn/ontap-mediator/index.html> on October 19, 2026. Always check docs.netapp.com for the latest.

目录

ONTAP Mediator 文档	1
发行说明	2
ONTAP Mediator 中的新增功能	2
ONTAP Mediator 支持矩阵	4
操作系统支持矩阵	4
SCST 支持矩阵	7
开始使用	9
了解 ONTAP Mediator	9
ONTAP Mediator 的工作原理	9
ONTAP 调解器服务	9
为系统管理员提供的工具	9
SCST 和内核更新	10
支持 HTTPS 和 iSCSI	10
ONTAP Mediator 快速入门	10
下一步是什么?	11
安装和设置	12
ONTAP Mediator 安装工作流程摘要	12
准备安装 ONTAP Mediator	12
安装要求	13
主机要求	13
防火墙和网络要求	14
OS 要求	14
启用 UEFI 安全启动时安装 ONTAP Mediator	15
为 ONTAP Mediator 安装提供存储库访问权限	16
下载 ONTAP Mediator 安装包	24
验证 ONTAP Mediator 代码签名	25
安装 ONTAP Mediator 安装包	27
注册 UEFI 安全启动的安全密钥	45
签名 SCST 内核模块	46
确认 ONTAP Mediator 安装状态	47
使用 ONTAP Mediator	49
使用 ONTAP 调解器进行 SnapMirror 活动同步和 MetroCluster IP	49
监控 SnapMirror 活动同步关系	49
监控 MetroCluster IP 配置	49
管理 ONTAP 调解器	50
升级 ONTAP Mediator	50
升级 ONTAP Mediator	50
升级主机操作系统并重新安装或升级 ONTAP Mediator	58
配置 ONTAP Mediator 属性	63

ONTAP Mediator 可配置设置	63
安装第三方 SSL 证书	63
配置密码攻击防护	65
配置密码复杂性规则	66
配置可用空间阈值	66
配置预留日志空间	67
更改 ONTAP Mediator 用户名和密码	67
更改用户名	67
更改密码	68
启动和停止 ONTAP Mediator	69
停止 ONTAP Mediator	69
启动 ONTAP Mediator	70
验证 ONTAP Mediator 运行状况	70
验证服务状态	70
确认侦听端口	71
管理证书	72
重新生成 ONTAP Mediator 主机的临时自签名证书	72
将 ONTAP Mediator 自签名证书替换为受信任的第三方证书	73
对 ONTAP Mediator 证书相关问题进行故障诊断	87
维护 ONTAP Mediator 的主机操作系统	92
重新启动主机	92
主机包更新	92
升级主机 OS 内核	92
执行主机维护	93
主机更改主机名或 IP	96
卸载 ONTAP Mediator	97
法律声明	98
版权	98
商标	98
专利	98
隐私政策	98
开源	98
用于 MetroCluster IP 和 SnapMirror 主动同步的 ONTAP 调解器	98

ONTAP Mediator 文档

发行说明

ONTAP Mediator 中的新增功能

ONTAP Mediator 的每个版本都提供了新的增强功能。



有关操作系统和 SCST 版本信息，请参见 ["支持矩阵"](#)。

ONTAP Mediator 版本	增强功能
1.12.1	• 支持 RHEL： ◦ 兼容：10.1 和 9.7。 ◦ 推荐：10.2、10.0、9.8、9.6、9.4 和 8.10。 • 支持 Rocky Linux 10.2、9.8 和 8.10。 • 支持使用 rhck 和 uek 内核的 Oracle Linux 10.1 和 9.8。 • 从 ONTAP Mediator 1.12.1 和 ONTAP 9.19.1 开始，配置了 HTTPS 的单个 ONTAP Mediator 实例可以同时监控以下配置： ◦ 如果安装在具有 8 核 CPU 和至少 16GB RAM 的主机上，则最多可安装 10 个四节点或 5 个八节点 MetroCluster IP 配置。请参阅 "使用 ONTAP Mediator 监控多个 MetroCluster IP 配置"。 ◦ 如果安装在具有双核 CPU 的主机上，则为 1 个四节点 MetroCluster IP 配置和 1 个 SnapMirror 活动同步对等关系。请参阅 "配置 ONTAP 调解器以进行 SnapMirror 活动同步"。 • 包括用于修复损坏的 SCST 安装的新工具。安装 ONTAP Mediator 后，您可以在 <code>/usr/local/bin/mediator_repair_scst</code> 找到该工具。如果升级后的操作系统需要默认 SCST 软件包中未提供的 SCST 修补程序，则可以在操作系统升级后使用该工具。
1.12	• 支持 RHEL： ◦ 兼容：10.1 和 9.7。 ◦ 推荐：10.2、10.0、9.8、9.6、9.4 和 8.10。 • 支持 Rocky Linux 10.2、9.8 和 8.10。 • 支持使用 rhck 和 uek 内核的 Oracle Linux 10.1 和 9.7。 • 从 ONTAP 9.19.1 和 ONTAP Mediator 1.12 开始，您可以在 MetroCluster IP 配置中配置使用 HTTPS 进行通信的 ONTAP Mediator。HTTPS 通过使用 SSL/TLS 加密整个会话来提高安全性。您可以选择安装仅支持 HTTPS 的 ONTAP Mediator。如果使用此安装方法，则不会安装 SCST 软件包。"详细了解 ONTAP Mediator 的 HTTPS 和 iSCSI 安装选项"。

1.11	• 支持 RHEL 10.1、10.0、9.7、9.6、9.4 和 8.10。 • 支持 Rocky Linux 10.1、9.7 和 8.10。 • 支持使用 rhck 和 uek 内核的 Oracle Linux 10.0 和 9.6。 • 为 MetroCluster IP 配置添加对 IPv6 的支持。 • 增加了对 fapolicyd 的支持。
1.10	• 支持 RHEL 10.0、9.6、9.4 和 8.10。 • 支持 Rocky Linux 10.0、9.6 和 8.10。 • 将基础 Python 版本从 Python 3.9 升级到 Python 3.12。
1.9.1	• 支持 RHEL： ◦ 兼容：9.3、9.1、8.9、8.7、8.6、8.5 和 8.4。 ◦ 推荐：9.5、9.4、9.2、9.0、8.10 和 8.8。 • 支持 Rocky Linux 9.5 和 8.10。 • 添加用于代码签名验证的新证书。 • 添加了对使用 <code>-skip-code-signature-check</code> 标志跳过代码签名检查的支持。 • 当安装程序检测到过期的代码签名证书时，会显示警告。
1.9	• 支持 RHEL： ◦ 兼容：9.3、9.1、8.9、8.7、8.6、8.5 和 8.4。 ◦ 推荐：9.5、9.4、9.2、9.0、8.10 和 8.8。 • 支持 Rocky Linux 9.5 和 8.10。 • 对 RHEL 和 Rocky Linux 的 FIPS 支持。 • 添加了性能增强功能，以实现更大的可扩展性。 • 改进了文件名，以简化 PKI 签名证书的设置。
1.8	• 支持 RHEL： ◦ 兼容：8.7、8.6、8.5 和 8.4。 ◦ 建议：9.4、9.3、9.2、9.1、9.0、8.10、8.9 和 8.8。 • 支持 Rocky Linux 9.4 和 8.10。
1.7	• 支持 RHEL： ◦ 兼容：8.7、8.6、8.5 和 8.4。 ◦ 推荐：9.3、9.2、9.1、9.0、8.9 和 8.8。 • 支持 Rocky Linux 9.3 和 8.9。 • 在自签名证书和第三方签名证书中支持 SAN（使用者备用名称）数据。

1.6	• Python 3.9 更新。 • 支持 RHEL： ◦ 兼容：8.7、8.6、8.5 和 8.4。 ◦ 建议：9.2、9.1、9.0 和 8.8。 • 支持 Rocky Linux 9.2 和 8.8。 • 停止支持 RHEL 7.x / CentOS 所有版本。
1.5	• 支持 RHEL 8.5、8.4、8.3、8.2、8.1、8.0、7.9、7.8、7.7 和 7.6。 • 支持 CentOS 7.9、7.8、7.7 和 7.6。 • 包括 RHEL 7.x / CentOS 7.x 的弃用警告。 • 为更大规模的 SnapMirror 主动同步系统优化速度。 • 已将加密代码签名添加到安装程序。
1.4	• 支持 RHEL 8.5、8.4、8.3、8.2、8.1、8.0、7.9、7.8、7.7 和 7.6。 • 支持 CentOS 7.9、7.8、7.7 和 7.6。 • 增加了对基于 UEFI 的固件的安全启动（SB）的支持。
1.3	• 支持 RHEL 8.3、8.2、8.1、8.0、7.9、7.8、7.7 和 7.6。 • 支持 CentOS 7.9、7.8、7.7 和 7.6。
1.2	• 支持 RHEL 8.1、8.0、7.9、7.8、7.7 和 7.6。 • 支持 CentOS 7.9、7.8、7.7 和 7.6。 • 支持 HTTPs 邮箱。 • 适用于 ONTAP 9.8+ MCC-IP AUSO 和 SnapMirror active sync ZRTO。
1.1	• 支持 RHEL 8.0 和 7.6。 • 支持 CentOS 7.6。 • 消除 Perl 依赖关系。
1.0	• 支持 iSCSI 邮箱。 • 适用于 ONTAP 9.7+ MCC-IP AUSO。 • 支持 RHEL/CentOS 7.6。

ONTAP Mediator 支持矩阵

ONTAP Mediator 支持矩阵包括操作系统和 SCST 版本支持信息。

操作系统支持矩阵

下表列出了每个版本的 ONTAP Mediator 支持的操作系统。

适用于 ONT AP Mediator 的操作系统	1.12.1	1.12	1.11	1.10	1.9.1	1.9	1.8	1.7	1.6	1.5	1.4	1.3	1.2	1.1	1.0
RHEL 10.2	是	是	否	否	否	否	否	否	否	否	否	否	否	否	否
RHEL 10.1	兼容	兼容	是	否	否	否	否	否	否	否	否	否	否	否	否
RHEL 10.0	是	是	是	是	否	否	否	否	否	否	否	否	否	否	否
RHEL 9.8	是	是	否	否	否	否	否	否	否	否	否	否	否	否	否
RHEL 9.7	兼容	兼容	是	否	否	否	否	否	否	否	否	否	否	否	否
RHEL 9.6	是	是	是	是	否	否	否	否	否	否	否	否	否	否	否
RHEL 9.5	否	否	否	否	是	否	否	否	否	否	否	否	否	否	否
RHEL 9.4	是	是	是	是	是	是	是	否	否	否	否	否	否	否	否
RHEL 9.3	否	否	否	否	否	否	是	是	否	否	否	否	否	否	否
RHEL 9.2	否	否	否	否	是	是	是	是	是	否	否	否	否	否	否
RHEL 9.1	否	否	否	否	否	否	是	是	是	否	否	否	否	否	否
RHEL 9.0	否	否	否	否	是	是	是	是	是	否	否	否	否	否	否
RHEL 8.10	是	是	是	是	是	是	是	否	否	否	否	否	否	否	否
RHEL 8.9	否	否	否	否	兼容	兼容	是	是	否	否	否	否	否	否	否
RHEL 8.8	否	否	否	否	是	是	是	是	是	否	否	否	否	否	否
RHEL 8.7	否	否	否	否	兼容	兼容	是	是	是	否	否	否	否	否	否

RHEL 8.6	否	否	否	否	兼容	兼容	是	是	是	否	否	否	否	否	否
RHEL 8.5	否	否	否	否	兼容	兼容	是	是	是	是	是	否	否	否	否
RHEL 8.4	否	否	否	否	兼容	兼容	是	是	是	是	是	否	否	否	否
RHEL 8.3	已废弃	已废弃	已废弃	已废弃	已废弃	已废弃	已废弃	已废弃	已废弃	是	是	是	否	否	否
RHEL 8.2	已废弃	已废弃	已废弃	已废弃	已废弃	已废弃	已废弃	已废弃	已废弃	是	是	是	否	否	否
RHEL 8.1	已废弃	已废弃	已废弃	已废弃	已废弃	已废弃	已废弃	已废弃	已废弃	是	是	是	是	否	否
RHEL 8.0	已废弃	已废弃	已废弃	已废弃	已废弃	已废弃	已废弃	已废弃	已废弃	是	是	是	是	是	否
RHEL 和 Cent OS 7.9	已废弃	已废弃	已废弃	已废弃	已废弃	已废弃	已废弃	已废弃	已废弃	是	是	是	兼容	否	否
RHEL 和 Cent OS 7.8	已废弃	已废弃	已废弃	已废弃	已废弃	已废弃	已废弃	已废弃	已废弃	是	是	是	是	否	否
RHEL 和 Cent OS 7.7	已废弃	已废弃	已废弃	已废弃	已废弃	已废弃	已废弃	已废弃	已废弃	是	是	是	是	否	否
RHEL 和 Cent OS 7.6	已废弃	已废弃	已废弃	已废弃	已废弃	已废弃	已废弃	已废弃	已废弃	是	是	是	是	是	是（仅限 RHEL）
Cent OS 8 和 stream	否	否	否	否	否	否	否	否	否	否	否	否	不适用	不适用	不适用
Rocky Linux 10	是 (10.2)	是 (10.2)	是 (10.0 和 10.1)	是	否	否	否	否	否	否	否	否	否	否	否
Rocky Linux 9	是 (9.8)	是 (9.8)	是 (9.7)	是 (9.6 和 9.4)	是 (9.5)	是 (9.5)	是 (9.4)	是 (9.3)	是 (9.2)	不适用	不适用	不适用	不适用	不适用	不适用

Rocky Linux 8	是 (8.10)	是 (8.10)	是 (8.10)	是 (8.10)	是 (8.10)	是 (8.10)	是 (8.10)	是 (8.9)	是 (8.8)	不适用	不适用	不适用	不适用	不适用	不适用
Oracle Linux 10	是 (10.1)	是 (10.1)	是 (10.0)	否	否	否	否	否	否	否	否	否	否	否	否
Oracle Linux 9	是 (9.8)	是 (9.7)	是 (9.6)	否	否	否	否	否	否	否	否	否	否	否	否

- “是”表示建议将该操作系统用于 ONTAP Mediator 安装，并且完全兼容且受支持。
- “否”表示操作系统和 ONTAP Mediator 不兼容。
- “兼容”意味着 Red Hat 不再支持这些 RHEL 版本，但仍可以在其上安装 ONTAP Mediator。
- ONTAP Mediator 1.6 增加了对 Rocky Linux 9 和 8 的支持。
- ONTAP Mediator 1.5 是 RHEL 7.x 分支操作系统的最后一个受支持版本。
- Centos 8 因其重新分支而从所有版本中被移除。Centos Stream 被认为不是合适的生产目标操作系统。没有计划提供支持。

SCST 支持矩阵

下表显示了每个版本的 ONTAP Mediator 支持的 SCST 版本。

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

ONTAP Mediator 版本	支持的 SCST 版本
ONTAP Mediator 1.0	scst-3.3.0.tar.bz2

开始使用

了解 ONTAP Mediator

ONTAP Mediator 支持 ONTAP 功能，为高可用性（HA）元数据提供持久且隔离的存储，提供同步节点运行状况查询功能以帮助确定仲裁，并充当控制器存活状态的 ping 代理。



本文档指的是本地版本的 ONTAP Mediator。有关从 ONTAP 9.17.1 开始提供的 ONTAP Cloud Mediator 的信息，请参见 ["SnapMirror 活动同步文档"](#)。

ONTAP Mediator 的工作原理

ONTAP Mediator 在第三故障域中的 Linux 主机上运行，与您的 ONTAP 集群分开，为 HA 元数据提供持久的隔离存储。ONTAP 使用此存储以及 Mediator 的节点运行状况查询和 ping 代理功能来监控 SnapMirror 活动同步关系和 MetroCluster IP 配置。

- ONTAP Mediator 使用节点管理 LIF 和集群管理 LIF 通过多个冗余路径与每个集群进行通信。这使得 ONTAP 能够区分整个站点故障和单个 InterSwitch 链路（ISL）故障。
- 在 SnapMirror 活动同步中，如果集群失去与其对等节点和 ONTAP Mediator 的连接，ONTAP 会将该对等节点视为无法访问，并可以自动故障转移到幸存的副本。
- 在 MetroCluster IP 配置中，ONTAP 调解器通过存储有关 MetroCluster 节点的状态信息并帮助确定故障期间的适当操作过程来协助调解器辅助的自动计划外切换（MAUSO）。

ONTAP 调解器服务

ONTAP Mediator 提供两项额外的 systemctl 服务：

- **ontap_mediator.service**

维护 REST API 服务器以管理 ONTAP 关系。

- **mediator-scst.service**

控制 iSCSI 模块（SCST）的启动和关闭。

为系统管理员提供的工具

为系统管理员提供的工具：

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

提供当前 API 用户名和密码时设置新 API 密码。

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

提供当前 API 用户名和密码时设置新的 API 用户名。

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

生成一个本地 tgz 文件，其中包含与 NetApp 客户支持进行通信所需的所有有用的支持信息。这包括应用程序配置、日志和一些系统信息。捆绑包在本地磁盘上生成，可以根据需要手动传输。存储位置：
： /opt/netapp/data/support_bundles/

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

删除 ONTAP Mediator 包和 SCST 内核模块。这包括所有配置、日志和邮箱数据。

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

如果达到身份验证重试限制，则解除对 API 用户帐户的锁定。此功能用于防止暴力破解密码。它会提示用户输入正确的用户名和密码。

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

(仅限支持人员) 用于在安装时添加 API 用户。

SCST 和内核更新

ONTAP Mediator 使用 SCST 为 MetroCluster IP 配置中的通信提供 iSCSI。SCST 包是在安装过程中编译的内核模块。更新内核后，可能需要重新安装 SCST。或者，卸载 ONTAP Mediator，然后重新安装并重新配置 ONTAP 关系。请参阅 ["SCST：适用于 Linux 的通用 SCSI 目标子系统"](#) 以了解有关 SCST 的更多信息。



您应该在 ONTAP 维护窗口中协调对服务器操作系统内核的任何更新。

支持 HTTPS 和 iSCSI

从 ONTAP 9.19.1 和 ONTAP Mediator 1.12 开始，您可以配置使用 HTTPS 或 iSCSI 监控 MetroCluster IP 配置的 ONTAP Mediator。您可以选择安装支持 iSCSI 和 HTTPS 或仅支持 HTTPS 的 ONTAP Mediator。如果安装 ONTAP Mediator 时仅支持 HTTPS，则不会安装 SCST 软件包。有关详细信息，请参见["适用于 ONTAP Mediator 的 HTTPS 和 iSCSI 安装选项"](#)。

ONTAP Mediator 快速入门

要开始使用 ONTAP Mediator，请安装该软件并将其配置为与您的 ONTAP 环境配合使用。

1

安装 ONTAP Mediator

"安装 ONTAP Mediator" 在您的系统上，并确保满足所有必需的依赖项。

2

验证 ONTAP Mediator 安装

"确认 ONTAP Mediator 安装成功"通过查看服务状态并检查日志中是否有任何错误或警告。

3

为您的 ONTAP 环境配置 ONTAP 调解器

根据您是否要["将 ONTAP 调解器用于 MetroCluster IP、SnapMirror 主动同步或两者"](#)，请执行相应的配置步骤，

以确保正确应用所有必需的设置。

下一步是什么？

您现在可以使用 ONTAP Mediator 来监控 MetroCluster IP 配置、SnapMirror 活动同步配置或两者。有关详细信息，请参见"[SnapMirror 活动同步文档](#)"和"[MetroCluster 文档](#)"。

安装和设置

ONTAP Mediator 安装工作流程摘要

安装 ONTAP Mediator 包括准备安装、提供对存储库的访问、下载安装包、验证代码签名以及安装 ONTAP Mediator 包。

1

"准备安装 ONTAP Mediator"

要安装 ONTAP Mediator，您必须确保满足所有先决条件。

2

"提供存储库访问"

您应该启用对存储库的访问权限，以便 ONTAP Mediator 可以在安装过程中访问所需的软件包。

3

"下载 ONTAP Mediator 安装包"

从 ONTAP Mediator 下载页面下载 ONTAP Mediator 安装包。

4

"验证 ONTAP Mediator 安装包的代码签名"

NetApp 建议在安装 ONTAP Mediator 安装包之前验证 ONTAP Mediator 代码签名。

5

"安装 ONTAP Mediator 安装包"

下载安装包并在主机上运行安装程序。

6

"验证 ONTAP Mediator 安装"

安装 ONTAP Mediator 后，请验证其是否正在成功运行。

准备安装 ONTAP Mediator

要安装 ONTAP Mediator，您需要满足所有先决条件，下载安装包，并在主机上运行安装程序。



- 要将 ONTAP Mediator 升级到新版本，请参阅 ["升级 ONTAP Mediator"](#)。
- 如果需要升级主机操作系统，并在此过程中重新安装或升级 ONTAP Mediator，请参阅["升级主机操作系统和 ONTAP Mediator"](#)。

安装要求

安装 ONTAP Mediator 之前，请查看以下要求。



- ONTAP Mediator 1.8 及更早版本与 Red Hat Enterprise Linux (RHEL) FIPS 模式不兼容，并会导致其无法成功安装。您可以使用 `fips-mode-setup --check` 命令检查是否启用了 FIPS 模式。您可以使用 `fips-mode-setup --disable` 命令禁用 FIPS 模式。禁用 FIPS 模式后重新启动，以成功安装 ONTAP Mediator 1.8 或更早版本。
- 如果安装失败，您可能需要尝试更新版本的 ONTAP Mediator。
- 从 2025 年 6 月 15 日起，您无法安装或升级 ONTAP Mediator 1.9 和 1.8，因为它们的代码签名证书已过期。如果安装失败，请改用 ONTAP Mediator 1.9.1 补丁版本。

- 如果安装 `yum-utils` 软件包，您可以使用 `needs-restarting` 命令。
- 从 ONTAP Mediator 1.11 开始，MetroCluster IP 配置支持 IPv6。

适用于 ONTAP Mediator 的 HTTPS 和 iSCSI 安装选项

在安装 ONTAP Mediator 之前，请查看以下信息：

- 在 ONTAP Mediator 中，SCST 包用于为 MetroCluster IP 配置中的通信提供 iSCSI。SCST 包是在安装过程中编译的内核模块。从 ONTAP 9.19.1 和 Mediator 1.12 开始，您可以配置使用 HTTPS 而不是 iSCSI 监控 MetroCluster IP 配置的 ONTAP Mediator。
- 从 ONTAP 9.19.1 和 ONTAP Mediator 1.12 开始，您可以选择安装支持 iSCSI 和 HTTPS 的 ONTAP Mediator，或仅支持 HTTPS。如果您选择安装仅支持 HTTPS 的 ONTAP Mediator，则不会安装 SCST 软件包。
- 如果您从安装了 SCST 的 ONTAP Mediator 1.11 或更早版本升级，并且在升级期间选择了 HTTPS only 选项，则会卸载 SCST 包，并且新的 ONTAP Mediator 版本不包含 SCST 包。
- 如果您计划使用 ONTAP Mediator 来监控多个 MetroCluster IP 配置，则支持取决于您的 Mediator 类型、ONTAP 版本和操作系统。有关详细信息，请参见["使用 ONTAP Mediator 监控多个 MetroCluster IP 配置"](#)。
- 仅对以下用例支持安装支持 HTTPS 的 ONTAP Mediator：
 - 您计划仅使用 ONTAP Mediator 来监控您的 SnapMirror 活动同步集群。
 - 您计划使用 HTTPS 配置 ONTAP Mediator 以监视一个或多个 MetroCluster IP 配置。
- 在以下情况下，必须安装支持 iSCSI 和 HTTPS 的 ONTAP Mediator：
 - 您计划使用 iSCSI 配置 ONTAP Mediator 以监视一个或多个 MetroCluster IP 配置。

主机要求

在安装 RHEL 或 Rocky Linux 并配置关联的存储库时，请遵循这些要求。



如果修改安装或配置过程，则可能需要执行其他步骤。

Linux 发行版要求

- 根据 Red Hat 的最佳实践安装 RHEL 或 Rocky Linux。由于 CentOS 8.x 已结束生命周期，因此不建议使用 CentOS 8.x 的兼容版本。

- 在安装 ONTAP Mediator 时，确保系统可以访问所需的存储库，以便安装程序可以检索并安装所有必需的软件依赖项。
- 要使 yum 安装程序能够在 RHEL 存储库中查找依赖软件，请在安装期间或之后使用有效的 Red Hat 订阅注册系统。



有关详细信息，请参见 Red Hat Subscription Manager 文档。

防火墙和网络要求

ONTAP Mediator 使用多个端口与特定服务进行通信。

- 如果您使用的是第三方防火墙：
 - 必须启用 HTTPS 访问。
 - 必须将其配置为允许在端口 31784 和 3260 上进行访问。

使用默认的 Red Hat 或 CentOS 防火墙时，会在 Mediator 安装过程中自动配置防火墙。

- 对于无法访问互联网的 Linux 主机，请确保在本地存储库中提供所需的软件包。

如果在 Linux 环境中使用链路聚合控制协议（LACP），请配置内核并将 `sysctl net.ipv4.conf.all.arp_ignore`` 设置为 ``2`。

下表列出了必须在防火墙中允许的端口：



- 仅在 MetroCluster IP 配置中需要 iSCSI 端口。
- 正常操作不需要 22/tcp 端口，但您可以暂时启用它进行维护，并在维护会话完成后禁用它。

端口/服务	源	方向	目标	目的
22/tcp	管理主机	入站	ONTAP 调解器	（可选）SSH / ONTAP Mediator 管理
31784/tcp	集群管理和节点管理 LIF	入站	ONTAP Mediator Web 服务器	（必需）REST API (HTTPS)
3260/tcp	节点管理 LIF	入站	ONTAP Mediator iSCSI 目标	（MetroCluster IP 配置需要）邮箱的 iSCSI 数据连接



对于使用 HTTPS 进行通信的 SnapMirror 活动同步或 MetroCluster IP ONTAP Mediator，ONTAP 不需要启用或连接端口 3260。

OS 要求

您的操作系统必须满足以下要求：

- 64 位物理安装或虚拟机
- 8 GB 内存

- 1 GB 磁盘空间（用于应用程序安装、服务器日志和数据库）
- 用户：root 访问

查看 ["操作系统支持矩阵"](#)，了解每个版本的 ONTAP Mediator 支持的操作系统。

操作系统所需软件包

ONTAP Mediator 需要以下软件包：



软件包由 ONTAP Mediator 安装程序预安装或自动安装。

所有 RHEL/CentOS 版本	RHEL 10.x / Rocky Linux 10 的附加软件包	RHEL 9.x / Rocky Linux 9 的附加软件包	RHEL 8.x / Rocky Linux 8 的附加软件包
• openssl • openssl-devel • kernel-devel-\$(uname -r) • gcc • make • libselinux-utils • 补丁 • bzip2 • perl-Data-Dumper • perl-ExtUtils-MakeMaker • efibootmgr • mokutil	• python3.12 • python3.12-devel	• elfutils-libelf-devel • policycoreutils-python-utils • python3 • python3-devel	• elfutils-libelf-devel • policycoreutils-python-utils • redhat-lsb-core • python39 • python39-devel

Mediator 安装包是一个自解压缩的压缩 tar 文件，其中包括：

- 包含无法从受支持版本的存储库获取的所有依赖项的 RPM 文件。
- 安装脚本。

建议使用有效的 SSL 证书。



- 内核版本必须与操作系统版本相匹配。
- 请勿将内核升级到超过您的 ONTAP Mediator 版本所支持的操作系统版本，因为经过测试的 SCST 模块可能无法正常工作。

启用 **UEFI** 安全启动时安装 **ONTAP Mediator**

ONTAP Mediator 可以安装在启用或未启用 UEFI 安全启动的系统上。

关于此任务

- 如果不需要，或者正在排除 ONTAP Mediator 安装问题，您可以选择在安装 ONTAP Mediator 之前禁用 UEFI 安全启动。从您的计算机设置中禁用 UEFI 安全启动选项。
- 如果您安装了仅支持 HTTPS 的 ONTAP Mediator 1.12 或更高版本，则未安装 SCST 包，您可以跳过此任务。



有关禁用 UEFI 安全启动的详细说明，请参见主机操作系统的文档。

要安装启用了 UEFI 安全启动的 ONTAP Mediator，您必须先注册安全密钥，然后才能启动服务。密钥在 SCST 安装的编译步骤中生成，并保存为机器上的私钥和公钥对。使用 `mokutil` 实用程序将公钥作为机器所有者密钥（MOK）添加到您的 UEFI 固件中，使系统能够信任并加载已签名的模块。将 `mokutil` 密码短语保存在安全的位置，因为在重新启动系统以激活 MOK 时需要使用它。

步骤

1. 检查您的系统是否已启用 UEFI 安全启动：

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

结果表明此系统是否启用了 UEFI 安全引导。

条件	转到.....
UEFI 安全启动已启用	
UEFI 安全启动已禁用	"为 ONTAP Mediator 安装提供存储库访问权限"



- 系统会提示您创建一个必须存储在安全位置的密码短语。您需要此密码短语才能在 UEFI 引导管理器中启用密钥。
- ONTAP Mediator 1.2.0 及更早版本不支持此模式。

2. 如果未安装 `mokutil` 实用程序，请运行以下命令：

```
yum install mokutil
```

下一步是什么？

["为 ONTAP Mediator 安装提供存储库访问权限"](#)

为 ONTAP Mediator 安装提供存储库访问权限

您应该启用对存储库的访问权限，以便 ONTAP Mediator 可以在安装过程中访问所需的软件包。

步骤

1. 确定必须访问的存储库，如下表所示：

如果您的操作系统是.....	您必须提供对这些存储库的访问权限.....
----------------	-----------------------

Oracle Linux 10.x	• ol-10-for-x86_64-baseos-rpms • ol-10-for-x86_64-appstream-rpms
Oracle Linux 9.x	• ol-9-for-x86_64-baseos-rpms • ol-9-for-x86_64-appstream-rpms
RHEL 10.x	• rhel-10-for-x86_64-baseos-rpms • rhel-10-for-x86_64-appstream-rpms
RHEL 9.x	• rhel-9-for-x86_64-baseos-rpms • rhel-9-for-x86_64-appstream-rpms
RHEL 8.x	• rhel-8-for-x86_64-baseos-rpms • rhel-8-for-x86_64-appstream-rpms
RHEL 7.x	• rhel-7-server-optional-rpms
CentOS 7.x	• C7.6.1810 - 基本存储库
Rocky Linux 10	• appstream • baseos
Rocky Linux 9	• appstream • baseos
Rocky Linux 8	• appstream • baseos

2. 使用以下过程之一启用对上面列出的存储库的访问，以便 ONTAP Mediator 可以在安装过程中访问所需的软件包。



如果 ONTAP Mediator 依赖于“额外”和“可选”存储库中的 Python 模块，则可能需要访问 `rhel-X-for-x86\_64-extras-rpms` 和 `rhel-X-for-x86\_64-optional-rpms` 文件。

▼ Oracle Linux 10.x 操作系统的程序

如果您的操作系统是 **Oracle Linux 10.x**，请使用此过程启用对存储库的访问：

步骤

- a. 订阅所需的存储库：

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

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

以下示例显示了此命令的执行情况：

```
[root@localhost ~]# subscription-manager repos --enable ol-10-for-x86_64-baseos-rpms
```

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

- b. 运行以下命令来列出已启用的存储库：

```
yum repolist
```

新订阅的存储库应显示在列表中。

▼ Oracle Linux 9.x 操作系统的程序

如果您的操作系统是 **Oracle Linux 9.x**，请使用此过程启用对存储库的访问：

步骤

- a. 订阅所需的存储库：

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

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

以下示例显示了此命令的执行情况：

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

- b. 运行以下命令来列出已启用的存储库：

```
yum repolist
```

新订阅的存储库应显示在列表中。

▼ RHEL 10.x 操作系统的步骤

如果您的操作系统是 **RHEL 10.x**，请使用此过程启用对存储库的访问：

步骤

- a. 订阅所需的存储库：

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

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

以下示例显示了此命令的执行情况：

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

- b. 运行以下命令来列出已启用的存储库：

```
yum repolist
```

新订阅的存储库应显示在列表中。

▼ RHEL 9.x 操作系统的步骤

如果您的操作系统是 **RHEL 9.x**，请使用此过程启用对存储库的访问：

步骤

- a. 订阅所需的存储库：

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

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

以下示例显示了此命令的执行情况：

```
[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.
```

b. 运行以下命令来列出已启用的存储库：

```
yum repolist
```

新订阅的存储库应显示在列表中。

▼ RHEL 8.x 操作系统的步骤

如果您的操作系统是 **RHEL 8.x**，请使用此过程启用对存储库的访问：

步骤

a. 订阅所需的存储库：

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

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

以下示例显示了此命令的执行情况：

```
[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.
```

b. 运行以下命令来列出已启用的存储库：

```
yum repolist
```

新订阅的存储库应显示在列表中。

▼ RHEL 7.x 操作系统的步骤

如果您的操作系统是 **RHEL 7.x**，请使用此过程启用对存储库的访问：

步骤

- a. 订阅所需的存储库：

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

以下示例显示了此命令的执行情况：

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

- b. 运行以下命令来列出已启用的存储库：

```
yum repolist
```

以下示例显示了此命令的执行情况。“rhel-7-server-optional-rpms”存储库应显示在列表中。

```
[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 ~]#
```

▼ CentOS 7.x 操作系统的步骤

如果您的操作系统是 **CentOS 7.x**，请使用此过程启用对存储库的访问：



以下示例显示了 CentOS 7.6 的存储库，可能不适用于其他 CentOS 版本。请为您的 CentOS 版本使用基础存储库。

步骤

- 添加 C7.6.1810 - Base 存储库。C7.6.1810 - Base vault 存储库包含 ONTAP Mediator 所需的“kernel-devel”软件包。
- 将以下行添加到 /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
```

- 运行以下命令来列出已启用的存储库：

```
yum repolist
```

以下示例显示了此命令的执行情况。CentOS-7.6.1810 - Base 存储库应显示在列表中。

```
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 ~]#
```

▼ Rocky Linux 10、9 或 8 操作系统的步骤

如果您的操作系统是 **Rocky Linux 10**、**Rocky Linux 9** 或 **Rocky Linux 8**，请使用此过程启用对存储库的访问：

步骤

a. 订阅所需的存储库:

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

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

b. 执行 `clean` 操作:

```
dnf clean all
```

c. 验证存储库列表:

```
dnf repolist
```

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

```
[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 ~]#
```

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

Rocky Linux 8 - BaseOS

下一步是什么？

["下载 ONTAP Mediator 安装包"](#)

下载 ONTAP Mediator 安装包

下载 ONTAP Mediator 安装软件包并进行安装。

步骤

1. 从 ONTAP Mediator 下载页面下载 ONTAP Mediator 安装包。

["ONTAP Mediator 下载页面"](#)

2. 将 Mediator 安装包放在当前工作目录中：

```
ls ontap-mediator-1.12.1.tgz
```

```
ontap-mediator-1.12.1.tgz
```



对于 ONTAP Mediator 1.4 及更早版本，安装程序名为 `ontap-mediator.o`。

如果您的系统没有互联网访问权限，请确保安装程序可以访问所需的软件包。

3. 如果需要，将 Mediator 安装包移动到安装目录。
4. 解压缩安装程序包：

```
tar xvfz ontap-mediator-1.12.1.tgz
```

```
ontap-mediator-1.12.1/
ontap-mediator-1.12.1/csc-prod-ONTAP-Mediator.pem
ontap-mediator-1.12.1/csc-prod-chain-ONTAP-Mediator.pem
ontap-mediator-1.12.1/tsa-prod-ONTAP-Mediator.pem
ontap-mediator-1.12.1/tsa-prod-chain-ONTAP-Mediator.pem
ontap-mediator-1.12.1/ONTAP-Mediator-production.pub
ontap-mediator-1.12.1/ontap-mediator-1.12.1
ontap-mediator-1.12.1/ontap-mediator-1.12.1.sig.tsr
ontap-mediator-1.12.1/ontap-mediator-1.12.1.tsr
ontap-mediator-1.12.1/ontap-mediator-1.12.1.sig
```

下一步是什么？
"验证 ONTAP Mediator 代码签名"

验证 ONTAP Mediator 代码签名

NetApp 建议在安装前验证 ONTAP Mediator 代码签名。此步骤为可选步骤。

开始之前

在验证 ONTAP Mediator 代码签名之前，请确保您的系统满足这些需求。

- 
- 从 2025 年 6 月 15 日起，您无法安装或升级到 ONTAP Mediator 1.9 和 1.8，因为代码签名验证证书已过期。相反，请安装或升级到 ONTAP Mediator 1.10 或更高版本。
 - 如果系统不符合以下要求，则无需执行验证流程，您可以直接前往 ["安装 ONTAP Mediator 安装包"](#)。
- openssl 版本 1.0.2 至 3.0 用于基本验证
 - openssl 版本 1.1.0 或更高版本，用于时间戳管理机构（TSA）操作
 - 用于 OCSP 验证的公共 Internet 访问

下载包包含以下文件：

文件	问题描述
ONTAP-Mediator-production.pub	用于验证签名的公钥
csc-prod-chain-ONTAP-Mediator.pem	公共认证 CA 信任链
csc-prod-ONTAP-Mediator.pem	用于生成密钥的证书
ontap-mediator-1.12.1	1.12 版本的产品安装可执行文件
ontap-mediator-1.12.1.sig	经过 SHA-256 哈希处理，然后使用 csc-prod 密钥进行 RSA 签名的安装程序签名
ontap-mediator-1.12.1.sig.tsr	OCSCP 用于安装程序签名的吊销请求
ontap-mediator-1.12.1.tsr	时间戳签名请求文件
tlsa-prod-ONTAP-Mediator.pem	TSR 的公共证书
tlsa-prod-chain-ONTAP-Mediator.pem	TSR 的公共证书 CA 链

步骤

1. 使用在线证书状态协议（OCSP）对 `csc-prod-ONTAP-Mediator.pem` 执行吊销检查。
 - a. 查找证书的 OCSP URL。开发人员证书可能不提供 URI：

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

- b. 为证书生成 OCSP 请求。

```
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. 连接到 OCSP 管理器以发送 OCSP 请求：

```
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. 针对本地主机验证 CSC 的信任链和到期日期：

```
openssl verify
```



来自 PATH 的 openssl 版本必须具有有效的 `cert.pem`（非自签名）。

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

如果代码签名检查证书或时间戳证书已过期或无效，则命令失败。下载较新版本的 ONTAP Mediator。

3. 使用关联证书验证 ontap-mediator-1.12.1.sig.tsr 和 ontap-mediator-1.12.1.tsr 文件：

OpenSSL 3.x

```
openssl ts -verify -data ontap-mediator-1.12.1.sig -in ontap-mediator-1.12.1.sig.tsr -CAfile tsa-prod-chain-ONTAP-Mediator.pem -untrusted tsa-prod-ONTAP-Mediator.pem
```

OpenSSL 1.x

```
openssl ts -verify -data ontap-mediator-1.12.1 -in ontap-mediator-1.12.1.tsr -CAfile tsa-prod-chain-ONTAP-Mediator.pem -partial_chain
```



.tsr 文件包含与安装程序和代码签名相关联的时间戳响应。处理确认时间戳具有 TSA 的有效签名，并且您的输入文件未更改。您的机器在本地执行验证。您无需访问 TSA 服务器。

4. 根据密钥验证签名：

```
openssl -dgst -verify
```

```
openssl dgst -sha256 -verify ONTAP-Mediator-production.pub -signature  
ontap-mediator-1.12.1.sig ontap-mediator-1.12.1
```

下一步是什么？

["安装 ONTAP Mediator 安装包"](#)

安装 ONTAP Mediator 安装包

要安装 ONTAP Mediator，必须获取安装包并在主机上运行安装程序。



要升级现有的 ONTAP Mediator 安装，请改为参见 ["升级 ONTAP Mediator"](#)。

关于此任务

- 从 ONTAP 9.19.1 和 ONTAP Mediator 1.12 开始，您可以选择安装支持 iSCSI 和 HTTPS 的 ONTAP Mediator，或仅支持 HTTPS。如果您选择安装仅支持 HTTPS 的 ONTAP Mediator，则不会安装 SCST 软件包。
- 如果您从安装了 SCST 的 ONTAP Mediator 1.11 或更早版本升级，并且在升级期间选择了 HTTPS only 选项，则会卸载 SCST 包，并且新的 ONTAP Mediator 版本不包含 SCST 包。
- 如果您计划使用 ONTAP Mediator 来监控多个 MetroCluster IP 配置，则支持取决于您的 Mediator 类型、ONTAP 版本和操作系统。有关详细信息，请参见["使用 ONTAP Mediator 监控多个 MetroCluster IP 配置"](#)。
- 仅对以下用例支持安装支持 HTTPS 的 ONTAP Mediator：
 - 您计划仅使用 ONTAP Mediator 来监控您的 SnapMirror 活动同步集群。
 - 您计划使用 HTTPS 配置 ONTAP Mediator 以监视一个或多个 MetroCluster IP 配置。
- 在以下情况下，必须安装支持 iSCSI 和 HTTPS 的 ONTAP Mediator：
 - 您计划使用 iSCSI 配置 ONTAP Mediator 以监视一个或多个 MetroCluster IP 配置。
- 在安装或升级到 ONTAP Mediator 1.12 或更高版本期间，安装程序会提示您响应以下内容：

```
ONTAP Mediator can be installed with iSCSI and HTTPS support, or HTTPS support  
only.
```

```
Do you wish to install ONTAP Mediator with iSCSI and HTTPS support?
```

```
Y(es)/n(o):
```

- 如果您计划使用配置了 iSCSI 的 ONTAP Mediator 来监控一个或多个 MetroCluster IP 配置，请进行响应 Yes。
- 如果您计划仅使用 ONTAP Mediator 进行 SnapMirror 活动同步，或者计划使用 HTTPS 配置 ONTAP Mediator 以监控一个或多个 MetroCluster IP 配置，请响应。 No

或者，您可以在 `./ontap-mediator-1.12.1/ontap-mediator-1.12.1` 命令中使用 `--https-only` 选项来仅指定 HTTPS。

- 对于自动化用例，您可以使用 `./ontap-mediator-1.12.1/ontap-mediator-1.12.1 -y` 命令跳过安装程序提示。

步骤

1. 运行安装程序并根据需要响应提示：

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

```
[root@mediator_host ~]# ./ontap-mediator-1.12.1/ontap-mediator-1.12.1
```



要跳过安装期间的签名检查，请使用

`./ontap-mediator-1.12.1/ontap-mediator-1.12.1 --skip-code-signature-check` 命令

安装程序会创建所需的帐户并安装所需的程序包。如果已经安装了 ONTAP Mediator，安装程序会提示您升级；有关特定于升级的步骤和示例，请参见 ["升级 ONTAP Mediator"](#)。

▼ 支持 HTTPS 和 iSCSI 的 ONTAP Mediator 安装示例（控制台输出）

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

ONTAP Mediator: Self Extracting Installer

```
+ Extracting the ONTAP Mediator installation/upgrade archive
+ Performing the ONTAP Mediator run-time code signature check
  Using openssl from the path: /usr/bin/openssl configured for
  CApath:/etc/pki/tls
+ Unpacking the ONTAP Mediator installer
+ ONTAP_MEDIATOR_INSTALL_HTTPS_ONLY is set to '0', args are: ''
+ Starting ONTAP Mediator install/update process: previous
  installation_mode is ''
```

ONTAP Mediator can be installed with iSCSI and HTTPS support, or support for HTTPS only.

Do you wish to install ONTAP Mediator with iSCSI and HTTPS support?

Y(es)/n(o): yes

ONTAP Mediator will be installed with mode: ISCSI_HTTPS

+ Selected install package mode: ISCSI_HTTPS

+ Checking that required ports 3260 and 31784 are free

ONTAP Mediator requires two user accounts. One for the service (netapp), and one for use by ONTAP to the mediator API (mediatoradmin).

Would you like to use the default account names: netapp + mediatoradmin? (Y(es)/n(o)): yes

Enter ONTAP Mediator user account (mediatoradmin) password:

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

+ Checking if SELinux is in enforcing mode

SELinux is set to Enforcing. ONTAP Mediator server requires modifying the SELinux context of the file

/opt/netapp/lib/ontap_mediator/pyenv/bin/uwsgi from type 'lib_t' to 'bin_t'.

This is necessary to start the ONTAP Mediator service while SELinux is set to Enforcing.

Allow SELinux context change? Y(es)/n(o): yes

The installer will change the SELinux context type of

/opt/netapp/lib/ontap_mediator/pyenv/bin/uwsgi from type 'lib_t' to 'bin_t'.

+ Checking for default Linux firewall

+ Installing required packages.

Last metadata expiration check: 2:16:00 ago on Tue 24 Mar 2026 12:58:28 PM EDT.

Package openssl-1:3.5.1-7.0.1.el9_7.x86_64 is already installed.

Package openssl-devel-1:3.5.1-7.0.1.el9_7.x86_64 is already installed.

Package kernel-uek-devel-6.12.0-109.67.6.el9uek.x86_64 is already installed.

Package gcc-11.5.0-11.0.2.el9.x86_64 is already installed.
 Package make-1:4.3-8.el9.x86_64 is already installed.
 Package which-2.21-30.el9_6.x86_64 is already installed.
 Package libselinux-utils-3.6-3.el9.x86_64 is already installed.
 Package patch-2.7.6-16.el9.x86_64 is already installed.
 Package perl-Data-Dumper-2.174-462.el9.x86_64 is already installed.
 Package perl-ExtUtils-MakeMaker-2:7.60-3.el9.noarch is already installed.
 Package bzip2-1.0.8-10.el9_5.x86_64 is already installed.
 Package efibootmgr-16-12.0.1.el9.x86_64 is already installed.
 Package mokutil-2:0.7.2-1.el9.x86_64 is already installed.
 Package python3-pip-21.3.1-1.el9.noarch is already installed.
 Package elfutils-libelf-devel-0.193-1.el9.x86_64 is already installed.
 Package policycoreutils-python-utils-3.6-3.el9.noarch is already installed.
 Package python3.12-3.12.12-4.0.1.el9_7.1.x86_64 is already installed.
 Package python3.12-devel-3.12.12-4.0.1.el9_7.1.x86_64 is already installed.
 Package gcc-toolset-14-14.0-2.el9.x86_64 is already installed.
 Dependencies resolved.

```

=====
=====
Package                               Architecture      Version
Repository                            Size
=====
=====
Installing:
  efibootmgr                          x86_64            18-
8.el10                                baseos            47 k
  mokutil                             x86_64
2:0.7.2-1.el10                        baseos            48 k
  patch                               x86_64            2.7.6-
26.el10                               appstream         131 k
  perl-ExtUtils-MakeMaker              noarch            2:7.70-
513.el10                              appstream         295 k
  python3-devel                        x86_64
3.12.12-4.el10_1                      appstream         329 k
  python3-pip                          noarch            23.3.2-
7.el10                               appstream         3.2 M
Upgrading:
  libselinux                          x86_64            3.9-
1.el10                                baseos            97 k
  libselinux-utils                    x86_64            3.9-
1.el10                                baseos            122 k
  libsemanage                         x86_64            3.9-
1.el10                                baseos            122 k

```

libsepol	x86_64	3.9-
1.el10	baseos	348 k
policycoreutils	x86_64	3.9-
1.el10	baseos	224 k
policycoreutils-python-utils	noarch	3.9-
1.el10	appstream	51 k
python-unversioned-command	noarch	
3.12.12-4.el10_1	appstream	11 k
python3	x86_64	
3.12.12-4.el10_1	baseos	28 k
python3-libs	x86_64	
3.12.12-4.el10_1	baseos	9.4 M
python3-libs-linux	x86_64	3.9-
1.el10	appstream	200 k
python3-libsemanage	x86_64	3.9-
1.el10	appstream	81 k
python3-policycoreutils	noarch	3.9-
1.el10	appstream	2.2 M
Installing dependencies:		
annobin-docs	noarch	12.99-
1.el10	appstream	88 k
annobin-plugin-gcc	x86_64	12.99-
1.el10	appstream	998 k
dwz	x86_64	0.16-
1.el10	appstream	139 k
efi-filesystem	noarch	6-
7.el10.rocky.0.1	baseos	7.8 k
efi-srpm-macros	noarch	6-
7.el10.rocky.0.1	appstream	22 k
efivar-libs	x86_64	39-
3.el10	baseos	127 k
fonts-srpm-macros	noarch	
1:2.0.5-18.el10	appstream	26 k
forge-srpm-macros	noarch	0.4.0-
6.el10	appstream	20 k
gcc-plugin-annobin	x86_64	14.3.1-
2.1.el10	appstream	67 k
go-srpm-macros	noarch	3.6.0-
7.el10_1	appstream	27 k
kernel-srpm-macros	noarch	1.0-
26.el10	appstream	9.4 k
lua-srpm-macros	noarch	1-
15.el10	appstream	8.8 k
ocaml-srpm-macros	noarch	10-
4.el10	appstream	9.2 k
openblas-srpm-macros	noarch	2-

19.el10	appstream	7.7 k
package-notes-srpm-macros	noarch	0.5-
13.el10	appstream	9.2 k
perl-AutoSplit	noarch	5.74-
512.2.el10_0	appstream	21 k
perl-Benchmark	noarch	1.25-
512.2.el10_0	appstream	27 k
perl-CPAN-Meta-Requirements	noarch	2.143-
11.el10	appstream	35 k
perl-CPAN-Meta-YAML	noarch	0.018-
512.el10	appstream	26 k
perl-Devel-PPPort	x86_64	3.72-
512.el10	appstream	219 k
perl-ExtUtils-Command	noarch	2:7.70-
513.el10	appstream	14 k
perl-ExtUtils-Constant	noarch	0.25-
512.2.el10_0	appstream	43 k
perl-ExtUtils-Install	noarch	2.22-
511.el10	appstream	43 k
perl-ExtUtils-Manifest	noarch	1:1.75-
511.el10	appstream	34 k
perl-ExtUtils-ParseXS	noarch	1:3.51-
512.el10	appstream	189 k
perl-File-Compare	noarch	
1.100.800-512.2.el10_0	appstream	13 k
perl-File-Copy	noarch	2.41-
512.2.el10_0	appstream	20 k
perl-I18N-Langinfo	x86_64	0.24-
512.2.el10_0	appstream	25 k
perl-JSON-PP	noarch	1:4.16-
512.el10	appstream	66 k
perl-Test-Harness	noarch	1:3.48-
512.el10	appstream	282 k
perl-lib	x86_64	0.65-
512.2.el10_0	appstream	15 k
perl-srpm-macros	noarch	1-
57.el10	appstream	8.5 k
perl-version	x86_64	
8:0.99.32-4.el10	appstream	63 k
pyproject-srpm-macros	noarch	1.16.2-
1.el10	appstream	14 k
python-srpm-macros	noarch	3.12-
10.el10	appstream	23 k
python3-pyparsing	noarch	3.1.1-
7.el10	baseos	271 k
qt6-srpm-macros	noarch	6.9.1-

1.el10	appstream	9.6 k
redhat-rpm-config	noarch	293-
1.el10.rocky.0.2	appstream	78 k
rust-toolset-srpm-macros	noarch	1.88.0-
1.el10	appstream	12 k
systemtap-sdt-devel	x86_64	5.3-
3b.el10	appstream	69 k
systemtap-sdt-dtrace	x86_64	5.3-
3b.el10	appstream	70 k
Installing weak dependencies:		
perl-CPAN-Meta	noarch	
2.150010-511.el10	appstream	197 k
perl-Encode-Locale	noarch	1.05-
31.el10	appstream	18 k
perl-Time-HiRes	x86_64	
4:1.9777-511.el10	appstream	58 k
perl-devel	x86_64	
4:5.40.2-512.2.el10_0	appstream	770 k
perl-doc	noarch	5.40.2-
512.2.el10_0	appstream	4.9 M

Transaction Summary

```
=====
=====
```

Install 52 Packages

Upgrade 12 Packages

Total download size: 26 M

Is this ok [y/N]: y

Downloading Packages:

(1/64): efi-filesystem-6-7.el10.rocky.0.1.noarch.rpm

206 kB/s | 7.8 kB 00:00

(2/64): efibootmgr-18-8.el10.x86_64.rpm

1.2 MB/s | 47 kB 00:00

(3/64): efivar-libs-39-3.el10.x86_64.rpm

2.8 MB/s | 127 kB 00:00

(4/64): mokutil-0.7.2-1.el10.x86_64.rpm

6.4 MB/s | 48 kB 00:00

(5/64): annobin-docs-12.99-1.el10.noarch.rpm

11 MB/s | 88 kB 00:00

(6/64): python3-pyparsing-3.1.1-7.el10.noarch.rpm

13 MB/s | 271 kB 00:00

(7/64): dwz-0.16-1.el10.x86_64.rpm

12 MB/s | 139 kB 00:00

(8/64): efi-srpm-macros-6-7.el10.rocky.0.1.noarch.rpm

2.7 MB/s | 22 kB 00:00

```
(9/64): fonts-srpm-macros-2.0.5-18.el10.noarch.rpm
5.1 MB/s | 26 kB      00:00
(10/64): forge-srpm-macros-0.4.0-6.el10.noarch.rpm
4.3 MB/s | 20 kB      00:00
(11/64): gcc-plugin-annobin-14.3.1-2.1.el10.x86_64.rpm
10 MB/s | 67 kB      00:00
(12/64): go-srpm-macros-3.6.0-7.el10_1.noarch.rpm
4.7 MB/s | 27 kB      00:00
(13/64): kernel-srpm-macros-1.0-26.el10.noarch.rpm
2.0 MB/s | 9.4 kB     00:00
(14/64): lua-srpm-macros-1-15.el10.noarch.rpm
2.0 MB/s | 8.8 kB     00:00
(15/64): ocaml-srpm-macros-10-4.el10.noarch.rpm
2.0 MB/s | 9.2 kB     00:00
(16/64): annobin-plugin-gcc-12.99-1.el10.x86_64.rpm
16 MB/s | 998 kB     00:00
(17/64): openblas-srpm-macros-2-19.el10.noarch.rpm
432 kB/s | 7.7 kB     00:00
(18/64): package-notes-srpm-macros-0.5-13.el10.noarch.rpm
519 kB/s | 9.2 kB     00:00
(19/64): perl-AutoSplit-5.74-512.2.el10_0.noarch.rpm
4.5 MB/s | 21 kB      00:00
(20/64): patch-2.7.6-26.el10.x86_64.rpm
14 MB/s | 131 kB     00:00
(21/64): perl-Benchmark-1.25-512.2.el10_0.noarch.rpm
3.5 MB/s | 27 kB      00:00
(22/64): perl-CPAN-Meta-Requirements-2.143-11.el10.noarch.rpm
10 MB/s | 35 kB      00:00
(23/64): perl-CPAN-Meta-YAML-0.018-512.el10.noarch.rpm
5.3 MB/s | 26 kB      00:00
(24/64): perl-CPAN-Meta-2.150010-511.el10.noarch.rpm
12 MB/s | 197 kB     00:00
(25/64): perl-Encode-Locale-1.05-31.el10.noarch.rpm
2.6 MB/s | 18 kB      00:00
(26/64): perl-ExtUtils-Command-7.70-513.el10.noarch.rpm
3.0 MB/s | 14 kB      00:00
(27/64): perl-Devel-PPPort-3.72-512.el10.x86_64.rpm
13 MB/s | 219 kB     00:00
(28/64): perl-ExtUtils-Constant-0.25-512.2.el10_0.noarch.rpm
4.9 MB/s | 43 kB      00:00
(29/64): perl-ExtUtils-Install-2.22-511.el10.noarch.rpm
4.7 MB/s | 43 kB      00:00
(30/64): perl-ExtUtils-Manifest-1.75-511.el10.noarch.rpm
6.2 MB/s | 34 kB      00:00
(31/64): perl-File-Compare-1.100.800-512.2.el10_0.noarch.rpm
4.2 MB/s | 13 kB      00:00
```

```
(32/64): perl-ExtUtils-MakeMaker-7.70-513.el10.noarch.rpm
17 MB/s | 295 kB      00:00
(33/64): perl-ExtUtils-ParseXS-3.51-512.el10.noarch.rpm
11 MB/s | 189 kB      00:00
(34/64): perl-File-Copy-2.41-512.2.el10_0.noarch.rpm
1.7 MB/s | 20 kB      00:00
(35/64): perl-Il8N-Langinfo-0.24-512.2.el10_0.x86_64.rpm
5.0 MB/s | 25 kB      00:00
(36/64): perl-JSON-PP-4.16-512.el10.noarch.rpm
9.3 MB/s | 66 kB      00:00
(37/64): perl-Time-HiRes-1.9777-511.el10.x86_64.rpm
11 MB/s | 58 kB      00:00
(38/64): perl-Test-Harness-3.48-512.el10.noarch.rpm
13 MB/s | 282 kB     00:00
(39/64): perl-lib-0.65-512.2.el10_0.x86_64.rpm
4.4 MB/s | 15 kB      00:00
(40/64): perl-srpm-macros-1-57.el10.noarch.rpm
2.7 MB/s | 8.5 kB     00:00
(41/64): perl-devel-5.40.2-512.2.el10_0.x86_64.rpm
19 MB/s | 770 kB     00:00
(42/64): perl-version-0.99.32-4.el10.x86_64.rpm
4.3 MB/s | 63 kB      00:00
(43/64): pyproject-srpm-macros-1.16.2-1.el10.noarch.rpm
4.5 MB/s | 14 kB      00:00
(44/64): python-srpm-macros-3.12-10.el10.noarch.rpm
5.1 MB/s | 23 kB      00:00
(45/64): python3-devel-3.12.12-4.el10_1.x86_64.rpm
20 MB/s | 329 kB     00:00
(46/64): qt6-srpm-macros-6.9.1-1.el10.noarch.rpm
3.0 MB/s | 9.6 kB     00:00
(47/64): redhat-rpm-config-293-1.el10.rocky.0.2.noarch.rpm
16 MB/s | 78 kB      00:00
(48/64): rust-toolset-srpm-macros-1.88.0-1.el10.noarch.rpm
3.6 MB/s | 12 kB     00:00
(49/64): systemtap-sdt-devel-5.3-3b.el10.x86_64.rpm
11 MB/s | 69 kB      00:00
(50/64): systemtap-sdt-dtrace-5.3-3b.el10.x86_64.rpm
16 MB/s | 70 kB      00:00
(51/64): libselinux-3.9-1.el10.x86_64.rpm
10 MB/s | 97 kB      00:00
(52/64): libselinux-utils-3.9-1.el10.x86_64.rpm
14 MB/s | 122 kB     00:00
(53/64): libsemanage-3.9-1.el10.x86_64.rpm
13 MB/s | 122 kB     00:00
(54/64): libsepol-3.9-1.el10.x86_64.rpm
18 MB/s | 348 kB     00:00
```

```

(55/64): python3-pip-23.3.2-7.el10.noarch.rpm
21 MB/s | 3.2 MB      00:00
(56/64): policycoreutils-3.9-1.el10.x86_64.rpm
4.3 MB/s | 224 kB     00:00
(57/64): python3-3.12.12-4.el10_1.x86_64.rpm
5.7 MB/s | 28 kB      00:00
(58/64): policycoreutils-python-utils-3.9-1.el10.noarch.rpm
6.7 MB/s | 51 kB      00:00
(59/64): perl-doc-5.40.2-512.2.el10_0.noarch.rpm
17 MB/s | 4.9 MB      00:00
(60/64): python-unversioned-command-3.12.12-4.el10_1.noarch.rpm
175 kB/s | 11 kB      00:00
(61/64): python3-libs-3.9-1.el10.x86_64.rpm
16 MB/s | 200 kB      00:00
(62/64): python3-libsemanage-3.9-1.el10.x86_64.rpm
7.0 MB/s | 81 kB      00:00
(63/64): python3-policycoreutils-3.9-1.el10.noarch.rpm
28 MB/s | 2.2 MB      00:00
(64/64): python3-libs-3.12.12-4.el10_1.x86_64.rpm
31 MB/s | 9.4 MB      00:00

```

```
-----
Total
```

```
19 MB/s | 26 MB      00:01
```

```
Running transaction check
```

```
Transaction check succeeded.
```

```
Running transaction test
```

```
Transaction test succeeded.
```

```
Running transaction
```

```
  Preparing          :
```

```
1/1
```

```
  Upgrading          : python3-libs-3.12.12-4.el10_1.x86_64
```

```
1/76
```

```
  Upgrading          : python-unversioned-command-3.12.12-4.el10_1.noarch
```

```
2/76
```

```
  Upgrading          : python3-3.12.12-4.el10_1.x86_64
```

```
3/76
```

```
  Upgrading          : libsepol-3.9-1.el10.x86_64
```

```
4/76
```

```
  Upgrading          : libselinux-3.9-1.el10.x86_64
```

```
5/76
```

```
  Running scriptlet: libselinux-3.9-1.el10.x86_64
```

```
5/76
```

```
  Installing         : perl-version-8:0.99.32-4.el10.x86_64
```

```
6/76
```

```

Installing      : perl-File-Copy-2.41-512.2.el10_0.noarch
7/76
Installing      : perl-CPAN-Meta-Requirements-2.143-11.el10.noarch
8/76
Upgrading       : libsemanage-3.9-1.el10.x86_64
9/76
Upgrading       : python3-libselenium-3.9-1.el10.x86_64
10/76
Installing      : perl-Time-HiRes-4:1.9777-511.el10.x86_64
11/76
Installing      : perl-JSON-PP-1:4.16-512.el10.noarch
12/76
Installing      : perl-File-Compare-1.100.800-512.2.el10_0.noarch
13/76
Installing      : perl-ExtUtils-ParseXS-1:3.51-512.el10.noarch
14/76
Installing      : efivar-libs-39-3.el10.x86_64
15/76
Upgrading       : python3-libsemanage-3.9-1.el10.x86_64
16/76
Installing      : perl-ExtUtils-Command-2:7.70-513.el10.noarch
17/76
Installing      : perl-ExtUtils-Manifest-1:1.75-511.el10.noarch
18/76
Upgrading       : libselenium-utils-3.9-1.el10.x86_64
19/76
Upgrading       : polycoreutils-3.9-1.el10.x86_64
20/76
Running scriptlet: polycoreutils-3.9-1.el10.x86_64
20/76
Upgrading       : python3-polycoreutils-3.9-1.el10.noarch
21/76
Installing      : python3-pyparsing-3.1.1-7.el10.noarch
22/76
Installing      : systemtap-sdt-dtrace-5.3-3b.el10.x86_64
23/76
Installing      : systemtap-sdt-devel-5.3-3b.el10.x86_64
24/76
Installing      : python3-pip-23.3.2-7.el10.noarch
25/76
Installing      : rust-toolset-srpm-macros-1.88.0-1.el10.noarch
26/76
Installing      : qt6-srpm-macros-6.9.1-1.el10.noarch
27/76
Installing      : pyproject-srpm-macros-1.16.2-1.el10.noarch
28/76

```

```
Installing      : perl-srpm-macros-1-57.el10.noarch
29/76
Installing      : perl-lib-0.65-512.2.el10_0.x86_64
30/76
Installing      : perl-doc-5.40.2-512.2.el10_0.noarch
31/76
Installing      : perl-I18N-Langinfo-0.24-512.2.el10_0.x86_64
32/76
Installing      : perl-Encode-Locale-1.05-31.el10.noarch
33/76
Installing      : perl-ExtUtils-Constant-0.25-512.2.el10_0.noarch
34/76
Installing      : perl-Devel-PPPort-3.72-512.el10.x86_64
35/76
Installing      : perl-CPAN-Meta-YAML-0.018-512.el10.noarch
36/76
Installing      : perl-CPAN-Meta-2.150010-511.el10.noarch
37/76
Installing      : perl-Benchmark-1.25-512.2.el10_0.noarch
38/76
Installing      : perl-Test-Harness-1:3.48-512.el10.noarch
39/76
Installing      : perl-AutoSplit-5.74-512.2.el10_0.noarch
40/76
Installing      : package-notes-srpm-macros-0.5-13.el10.noarch
41/76
Installing      : openblas-srpm-macros-2-19.el10.noarch
42/76
Installing      : ocaml-srpm-macros-10-4.el10.noarch
43/76
Installing      : lua-srpm-macros-1-15.el10.noarch
44/76
Installing      : kernel-srpm-macros-1.0-26.el10.noarch
45/76
Installing      : gcc-plugin-annobin-14.3.1-2.1.el10.x86_64
46/76
Installing      : efi-srpm-macros-6-7.el10.rocky.0.1.noarch
47/76
Installing      : dwz-0.16-1.el10.x86_64
48/76
Installing      : annobin-docs-12.99-1.el10.noarch
49/76
Installing      : annobin-plugin-gcc-12.99-1.el10.x86_64
50/76
Installing      : fonts-srpm-macros-1:2.0.5-18.el10.noarch
51/76
```

```

Installing      : forge-srpm-macros-0.4.0-6.el10.noarch
52/76
Installing      : go-srpm-macros-3.6.0-7.el10_1.noarch
53/76
Installing      : python-srpm-macros-3.12-10.el10.noarch
54/76
Installing      : redhat-rpm-config-293-1.el10.rocky.0.2.noarch
55/76
Running scriptlet: redhat-rpm-config-293-1.el10.rocky.0.2.noarch
55/76
Installing      : perl-ExtUtils-Install-2.22-511.el10.noarch
56/76
Installing      : perl-devel-4:5.40.2-512.2.el10_0.x86_64
57/76
Installing      : perl-ExtUtils-MakeMaker-2:7.70-513.el10.noarch
58/76
Installing      : efi-filesystem-6-7.el10.rocky.0.1.noarch
59/76
Installing      : efibootmgr-18-8.el10.x86_64
60/76
Installing      : python3-devel-3.12.12-4.el10_1.x86_64
61/76
Upgrading       : policycoreutils-python-utils-3.9-1.el10.noarch
62/76
Installing      : mokutil-2:0.7.2-1.el10.x86_64
63/76
Installing      : patch-2.7.6-26.el10.x86_64
64/76
Cleanup         : policycoreutils-python-utils-3.8-1.el10.noarch
65/76
Cleanup         : python3-policycoreutils-3.8-1.el10.noarch
66/76
Running scriptlet: policycoreutils-3.8-1.el10.x86_64
67/76
Cleanup         : policycoreutils-3.8-1.el10.x86_64
67/76
Cleanup         : python3-libsemanage-3.8.1-1.el10_0.x86_64
68/76
Cleanup         : libsemanage-3.8.1-1.el10_0.x86_64
69/76
Cleanup         : python3-libselinux-3.8-2.el10_0.x86_64
70/76
Cleanup         : libselinux-utils-3.8-2.el10_0.x86_64
71/76
Cleanup         : python3-3.12.9-2.el10_0.2.x86_64
72/76

```

```

Cleanup          : libselinux-3.8-2.el10_0.x86_64
73/76
Cleanup          : python-unversioned-command-3.12.9-
2.el10_0.2.noarch
74/76
Cleanup          : libsepol-3.8-1.el10.x86_64
75/76
Cleanup          : python3-libs-3.12.9-2.el10_0.2.x86_64
76/76
Running scriptlet: python3-libs-3.12.9-2.el10_0.2.x86_64
76/76

Upgraded:
libselinux-3.9-1.el10.x86_64
libselinux-utils-3.9-1.el10.x86_64
libsemanage-3.9-1.el10.x86_64
libsepol-3.9-1.el10.x86_64
policycoreutils-3.9-1.el10.x86_64
policycoreutils-python-utils-3.9-1.el10.noarch
python-unversioned-command-3.12.12-4.el10_1.noarch
python3-3.12.12-4.el10_1.x86_64
python3-libs-3.12.12-4.el10_1.x86_64
python3-libselinux-3.9-1.el10.x86_64
python3-libsemanage-3.9-1.el10.x86_64
python3-policycoreutils-3.9-1.el10.noarch
Installed:
annobin-docs-12.99-1.el10.noarch
annobin-plugin-gcc-12.99-1.el10.x86_64
dwz-0.16-1.el10.x86_64
efi-
filesystem-6-7.el10.rocky.0.1.noarch
efi-srpm-macros-6-7.el10.rocky.0.1.noarch
efibootmgr-18-8.el10.x86_64
efivar-libs-39-3.el10.x86_64
fonts-srpm-macros-1:2.0.5-18.el10.noarch
forge-srpm-macros-0.4.0-6.el10.noarch
gcc-
plugin-annobin-14.3.1-2.1.el10.x86_64
go-srpm-macros-3.6.0-7.el10_1.noarch
kernel-srpm-macros-1.0-26.el10.noarch
lua-srpm-macros-1-15.el10.noarch
mokutil-2:0.7.2-1.el10.x86_64
ocaml-srpm-macros-10-4.el10.noarch
openblas-srpm-macros-2-19.el10.noarch
package-notes-srpm-macros-0.5-13.el10.noarch
patch-2.7.6-26.el10.x86_64
perl-AutoSplit-5.74-512.2.el10_0.noarch
perl-Benchmark-1.25-512.2.el10_0.noarch

```

```

perl-CPAN-Meta-2.150010-511.el10.noarch
perl-CPAN-Meta-Requirements-2.143-11.el10.noarch
perl-CPAN-Meta-YAML-0.018-512.el10.noarch
perl-Devel-PPPort-3.72-512.el10.x86_64
perl-Encode-Locale-1.05-31.el10.noarch
perl-ExtUtils-Command-2:7.70-513.el10.noarch
perl-ExtUtils-Constant-0.25-512.2.el10_0.noarch
perl-ExtUtils-Install-2.22-511.el10.noarch
perl-ExtUtils-MakeMaker-2:7.70-513.el10.noarch
perl-ExtUtils-Manifest-1:1.75-511.el10.noarch
perl-ExtUtils-ParseXS-1:3.51-512.el10.noarch
perl-File-Compare-1.100.800-512.2.el10_0.noarch
perl-File-Copy-2.41-512.2.el10_0.noarch
perl-I18N-Langinfo-0.24-512.2.el10_0.x86_64
perl-JSON-PP-1:4.16-512.el10.noarch
perl-Test-Harness-1:3.48-512.el10.noarch
perl-Time-HiRes-4:1.9777-511.el10.x86_64
perl-devel-4:5.40.2-512.2.el10_0.x86_64
perl-doc-5.40.2-512.2.el10_0.noarch
perl-lib-0.65-512.2.el10_0.x86_64
perl-srpm-macros-1-57.el10.noarch
perl-version-8:0.99.32-4.el10.x86_64
pyproject-srpm-macros-1.16.2-1.el10.noarch
python-srpm-macros-3.12-10.el10.noarch
python3-devel-3.12.12-4.el10_1.x86_64
python3-pip-23.3.2-7.el10.noarch
python3-pyparsing-3.1.1-7.el10.noarch
srpm-macros-6.9.1-1.el10.noarch
redhat-rpm-config-293-1.el10.rocky.0.2.noarch
rust-toolset-srpm-macros-1.88.0-1.el10.noarch
systemtap-sdt-devel-5.3-3b.el10.x86_64
systemtap-sdt-dtrace-5.3-3b.el10.x86_64

```

qt6-

Complete!

OS package installations finished

OS package installations finished

+ Installing ONTAP Mediator. (Log:

/root/ontap_mediator.pJBcIv/ontap-mediator-1.12.1/ontap-mediator-1.12.1/install_20260324151410.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_20260324151410.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 needed.

SCST will not start automatically when Secure Boot is enabled and not configured properly.

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

For more information, see
 /opt/netapp/lib/ontap_mediator/README

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

For more information, see
 /opt/netapp/lib/ontap_mediator/README

```
[root@mediator_host ~]#
```

▼ 仅支持 HTTPS 的 ONTAP Mediator 安装示例（控制台输出）

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

--https-only

ONTAP Mediator: Self Extracting Installer

```
+ Extracting the ONTAP Mediator installation/upgrade archive
+ Performing the ONTAP Mediator run-time code signature check
  Using openssl from the path: /usr/bin/openssl configured for
CApath:/etc/pki/tls
Error querying OCSP responder
40E72F35407F0000:error:1E800080:HTTP
routines:OSSL_HTTP_REQ_CTX_nbio:failed reading
data:crypto/http/http_client.c:596:
40E72F35407F0000:error:1E800067:HTTP
routines:OSSL_HTTP_REQ_CTX_exchange:error
receiving:crypto/http/http_client.c:1045:server=http://ocsp.entrust.net:80
```

WARNING: The OCSP check failed while attempting to test the Code-Signature-Check certificate

Continue without code signature checking (only recommended if integrity has been established manually)? y(es)/N(o): yes
SKIPPING: Code signature check, manual override due to lack of OCSP response

```
+ Unpacking the ONTAP Mediator installer
+ ONTAP_MEDIATOR_INSTALL_HTTPS_ONLY is set to '1', args are: ''
+ Starting ONTAP Mediator install/update process: previous
installation_mode is ''
+ Selected install package mode: HTTPS_ONLY
```

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

Enter ONTAP Mediator user account (mediatoradmin) password:

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

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

+ Checking for default Linux firewall

+ Installing required packages.

Last metadata expiration check: 0:38:52 ago on Mon 20 Apr 2026
11:23:22 AM EDT.

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

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

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

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

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

Package libselinux-utils-3.6-3.el9.x86_64 is already installed.

Package patch-2.7.6-16.el9.x86_64 is already installed.

Package perl-Data-Dumper-2.174-462.el9.x86_64 is already installed.

Package perl-ExtUtils-MakeMaker-2:7.60-3.el9.noarch is already
installed.

Package bzip2-1.0.8-10.el9_5.x86_64 is already installed.

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

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

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

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

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

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

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

Dependencies resolved.

Nothing to do.

Complete!

OS package installations finished

+ Installing ONTAP Mediator. (Log: /root/ontap_mediator.CJqNqu/ontap-
mediator-1.12.1/ontap-mediator-1.12.1/install_20260420120204.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_20260420120204.log)

+ Note: ONTAP Mediator generated a self-signed server certificate for
temporary use on

this host. If the DNS name or IP address for the host is changed,

```
the certificate
    will no longer be valid. The default certificates should be
replaced with secure
    trusted certificates signed by a known certificate authority prior
to use for production.
    For more information, see /opt/netapp/lib/ontap_mediator/README
[root@mediator_host ~]#
```

注册 UEFI 安全启动的安全密钥

从 ONTAP Mediator 1.4 开始，UEFI 系统上启用了安全启动机制。启用安全启动后，您必须在安装后执行额外步骤来注册安全密钥。



如果您安装了仅支持 HTTPS 的 ONTAP Mediator 1.12 或更高版本，则未安装 SCST 包，您可以跳过此任务。

步骤

1. 按照 README 文件中的说明对 SCST 内核模块进行签名：

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

2. 找到所需的密钥：

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



安装后，系统输出将提供 README 文件和密钥位置。

3. 请将公钥添加到 MOK 列表：

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



您可以将私钥保留在默认位置，也可以将其移动到安全位置。您必须将公钥保存在其现有位置，以便引导管理器可以使用它。有关详细信息，请参见 README.module-signing 文件：

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

4. 重启主机，并使用设备的 UEFI Boot Manager 批准新的 MOK。您需要为 mokutil 实用程序提供的密码短语（见 ["启用 UEFI 安全启动时安装 ONTAP Mediator"](#)）。

签名 SCST 内核模块

安装 ONTAP Mediator 后，如果 `systemctl status mediator-scst` 显示为失败（非活动），请按照以下步骤对 SCST 内核模块进行签名。



如果您安装了仅支持 HTTPS 的 ONTAP Mediator 1.12 或更高版本，则未安装 SCST 包，您可以跳过此任务。

步骤

1. 在构建过程中，会在 `/opt/netapp/lib/ontap_mediator/ontap_mediator/SCST_mod_keys/` 目录中生成公钥/私钥对。运行以下命令来查看生成的密钥：

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

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

2. 通过运行以下命令开始将公钥导入 UEFI 密钥存储库的过程：

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

`mokutil` 实用程序将提示您输入两次密码。

3. `mokutil` 软件请求在导入过程中对此密钥使用临时密码。
4. 使用以下命令验证导入过程是否已启动，然后重新启动系统。引导加载程序启动 EFI MOK 管理器。

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

5. 使用屏幕上的菜单打开 SCST 内核模块密钥。启动后，运行以下命令。服务启动后，将对 SCST 内核模块进行签名。

```
systemctl status mediator-scst
```

下一步是什么？

["确认 ONTAP Mediator 安装状态"](#)

确认 ONTAP Mediator 安装状态

安装 ONTAP Mediator 后，请验证其是否正在成功运行。

步骤

1. 查看 ONTAP Mediator 的状态：

```
systemctl status ontap_mediator
```

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

2. 查看 SCST 状态：

```
systemctl status mediator-scst
```



如果您安装了仅支持 HTTPS 的 ONTAP Mediator 1.12 或更高版本，则未安装 SCST 包，您可以跳过此步骤。

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

```
Tasks: 1 (limit: 49473)
Memory: 1.2M
CGroup: /system.slice/mediator-scst.service
└─286662 /usr/local/sbin/iscsi-scstd
```

3. 确认 ONTAP Mediator 使用的端口：

```
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
```

下一步是什么？

安装并验证 ONTAP Mediator 后，您可以继续执行以下安装后配置任务：

- ["使用 ONTAP 调解器进行 SnapMirror 活动同步和 MetroCluster IP"](#)
- ["配置 ONTAP Mediator 属性"](#)（可选）

使用 ONTAP Mediator

使用 ONTAP 调解器进行 SnapMirror 活动同步和 MetroCluster IP

ONTAP Mediator 在第三故障域中的 Linux 主机上运行，与您的 ONTAP 集群分开，为高可用性（HA）元数据提供持久的隔离存储。ONTAP 使用此存储以及 Mediator 的节点健康查询和 ping 代理功能，来监控 SnapMirror 活动同步关系和 MetroCluster IP 配置。



本文档指的是本地版本的 ONTAP Mediator。有关从 ONTAP 9.17.1 开始提供的 ONTAP Cloud Mediator 的信息，请参见 ["SnapMirror 活动同步文档"](#)。

监控 SnapMirror 活动同步关系

SnapMirror 主动同步使用 ONTAP Mediator 作为一致性组中主副本和镜像副本的被动见证。如果发生网络分区或某个副本不可用，SnapMirror 主动同步将查询 ONTAP Mediator 以确定哪个副本继续提供 I/O 服务，并停止另一个副本上的 I/O。

从 ONTAP 9.8 开始，您可以使用任何版本的 ONTAP Mediator 来监控 SnapMirror 活动同步关系。



有关单个 ONTAP Mediator 实例可以监视的 SnapMirror 活动同步群集对数量的限制，请参见 ["ONTAP SnapMirror 活动同步的对象限制"](#)。

["配置 ONTAP 调解器以进行 SnapMirror 活动同步"](#)

监控 MetroCluster IP 配置

在 MetroCluster IP 配置中，ONTAP 调解器在邮箱 LUN 中存储有关 MetroCluster 节点的状态信息，并在发生灾难或特定故障情形时协助节点进行调解器辅助的自动计划外切换（MAUSO）。

从 ONTAP 9.19.1 和 ONTAP Mediator 1.12 开始，您可以配置使用 HTTPS 进行通信的 ONTAP Mediator 来监控您的 MetroCluster IP 配置。当您使用 iSCSI 协议配置 ONTAP Mediator 时，您可以使用任何版本的 ONTAP Mediator 来监控 MetroCluster IP 配置。



有关单个 ONTAP Mediator 实例可以监控的 MetroCluster IP 配置数量的限制，请参阅 ["使用 ONTAP Mediator 监控多个 MetroCluster IP 配置"](#)

["为 MetroCluster IP 配置配置 ONTAP 调解器"](#)

相关信息

- ["ONTAP SnapMirror 主动同步架构"](#)
- ["了解 ONTAP Mediator 对 MetroCluster IP 配置中自动计划外切换的支持"](#)

管理 ONTAP 调解器

升级 ONTAP Mediator

升级 ONTAP Mediator

要将 ONTAP Mediator 升级到较新版本而不升级主机操作系统，请下载新版本的安装包，并在已安装 ONTAP Mediator 的主机上运行安装程序。



如果您需要将主机操作系统升级到新版本，并作为该过程的一部分重新安装或升级 ONTAP Mediator，请改为参见 ["升级主机操作系统和 ONTAP Mediator"](#)。

开始之前

- 确认您的主机满足["准备安装 ONTAP Mediator"](#)中所述的先决条件，包括主机、防火墙和操作系统要求。
- 下载新版本的安装软件包。请参阅 ["下载 ONTAP Mediator 安装包"](#)。
- NetApp 建议验证新安装包的代码签名。请参阅 ["验证 ONTAP Mediator 安装包的代码签名"](#)。

关于此任务

在升级 ONTAP Mediator 之前，请查看下列信息。

- 您应该将 ONTAP Mediator 升级到最新版本。旧版本仍然适用于所有 ONTAP 版本，但新版本包含第三方组件的安全补丁。
- 当您升级到包含 SCST 包的新 ONTAP Mediator 版本时，安装程序会自动升级到推荐的 SCST 版本，除非有更高版本可用。有关手动安装更高 SCST 版本的说明，请参见 ["维护 ONTAP Mediator 的主机操作系统"](#)。有关支持的版本，请参见 ["SCST 支持矩阵"](#)。



- 如果升级失败，您可能需要尝试更新版本的 ONTAP Mediator。

- 从 2025 年 6 月 15 日起，您无法安装或升级 ONTAP Mediator 1.9 和 1.8，因为它们的代码签名证书已过期。如果升级失败，请改用 ONTAP Mediator 1.9.1 补丁版本。

- 从 ONTAP 9.19.1 和 ONTAP Mediator 1.12 开始，您可以选择安装支持 iSCSI 和 HTTPS 的 ONTAP Mediator，或仅支持 HTTPS。如果您选择安装仅支持 HTTPS 的 ONTAP Mediator，则不会安装 SCST 软件包。
- 如果您从安装了 SCST 的 ONTAP Mediator 1.11 或更早版本升级，并且在升级期间选择了 HTTPS only 选项，则会卸载 SCST 包，并且新的 ONTAP Mediator 版本不包含 SCST 包。
- 如果您计划使用 ONTAP Mediator 来监控多个 MetroCluster IP 配置，则支持取决于您的 Mediator 类型、ONTAP 版本和操作系统。有关详细信息，请参见["使用 ONTAP Mediator 监控多个 MetroCluster IP 配置"](#)。
- 仅对以下用例支持安装支持 HTTPS 的 ONTAP Mediator：
 - 您计划仅使用 ONTAP Mediator 来监控您的 SnapMirror 活动同步集群。
 - 您计划使用 HTTPS 配置 ONTAP Mediator 以监视一个或多个 MetroCluster IP 配置。
- 在以下情况下，必须安装支持 iSCSI 和 HTTPS 的 ONTAP Mediator：

◦ 您计划使用 iSCSI 配置 ONTAP Mediator 以监视一个或多个 MetroCluster IP 配置。

- 当您针对现有的 ONTAP Mediator 安装运行安装程序时，它会检测到当前版本并提示您确认升级：

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

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

- 作为升级的一部分，安装程序会收集升级前配置的支持包，并在升级 ONTAP Mediator 软件之前迁移数据库。
- 对于自动化用例，您可以使用 `./ontap-mediator-1.12.1/ontap-mediator-1.12.1 -y` 命令跳过安装程序提示。

步骤

1. 运行新版本的安装程序并根据需要响应提示：

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



要跳过升级期间的签名检查，请使用

`./ontap-mediator-1.12.1/ontap-mediator-1.12.1 --skip-code-signature-check` 命令

安装程序检测到现有安装并提示您升级，然后安装所需的程序包并迁移配置。

▼ 支持 HTTPS 和 iSCSI 的 ONTAP Mediator 升级示例（控制台输出）

```
[root@mediator_host ~]# ontap-mediator-1.11.0/ontap-mediator-1.11.0

ONTAP Mediator: Self Extracting Installer

+ Extracting the ONTAP Mediator installation/upgrade archive
+ Performing the ONTAP Mediator run-time code signature check
  Using openssl from the path: /usr/bin/openssl configured for
  CApath:/etc/pki/tls
+ Unpacking the ONTAP Mediator installer
ONTAP Mediator requires two user accounts. One for the service
(netapp), and one for use by ONTAP to the mediator API
(mediatoradmin).
Using default account names: netapp + mediatoradmin

Enter ONTAP Mediator user account (mediatoradmin) password:

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

+ Checking if SELinux is in enforcing mode
```

+ Checking for default Linux firewall

+ Installing required packages.

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

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

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

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

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

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

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

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

Package libselinux-utils-3.6-3.el9.x86_64 is already installed.

Package patch-2.7.6-16.el9.x86_64 is already installed.

Package perl-Data-Dumper-2.174-462.el9.x86_64 is already installed.

Package perl-ExtUtils-MakeMaker-2:7.60-3.el9.noarch is already
installed.

Package bzip2-1.0.8-10.el9_5.x86_64 is already installed.

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

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

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

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

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

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

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

Dependencies resolved.

Nothing to do.

Complete!

OS package installations finished

+ Installing ONTAP Mediator. (Log: /root/ontap_mediator.uigddi/ontap-
mediator-1.11.0/ontap-mediator-1.11.0/install_20260420145003.log)

This step will take several minutes. Use the log file to view
progress.

Sudoer config verified

ONTAP Mediator rsyslog and logging rotation enabled

+ Install successful. (Moving log to

```
/opt/netapp/lib/ontap_mediator/log/install_20260420145003.log)
```

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

```
+ Note: ONTAP Mediator uses a kernel module compiled specifically for
the current
    OS. Using 'yum update' to upgrade the kernel might cause
service interruption.
    For more information, see /opt/netapp/lib/ontap_mediator/README
[root@mediator_host ~]# ontap-mediator-1.12.1/ontap-mediator-1.12.1
```

ONTAP Mediator: Self Extracting Installer

```
+ Extracting the ONTAP Mediator installation/upgrade archive
+ Performing the ONTAP Mediator run-time code signature check
    Using openssl from the path: /usr/bin/openssl configured for
CApath:/etc/pki/tls
+ Unpacking the ONTAP Mediator installer
+ ONTAP_MEDIATOR_INSTALL_HTTPS_ONLY is set to '0', args are: ''
+ Starting ONTAP Mediator install/update process: previous
installation_mode is ''
```

```
ONTAP Mediator can be installed with iSCSI and HTTPS support, or
support for HTTPS only.
Do you wish to install ONTAP Mediator with iSCSI and HTTPS support?
Y(es)/n(o): yes
ONTAP Mediator will be installed with mode: ISCSI_HTTPS
+ Selected install package mode: ISCSI_HTTPS
ONTAP Mediator will be upgraded from version 1.11.0 to 1.12.1.
```

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

```

Creating tar archive...
Support bundle has been generated at
/opt/netapp/data/support_bundles/scs001097036-1_1776711598_BASIC.tgz
Testing the DB can be upgraded
Cloning database...
Cloning successful.
Verifying database is up-to-date...
Database out-of-date.Migrations are necessary:
Current revision: 0d5dcf349b27, Head revision: 758e4873c40d
Performing database migrations against clone...
Migrations successful.
Database upgrade operation complete.
+ Adding python312 specific packages
+ Upgrading ONTAP Mediator. (Log: /root/ontap_mediator.gwaGmU/ontap-
mediator-1.12.1/ontap-mediator-1.12.1/upgrade_20260420145942.log)

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

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

```

▼ 仅支持 HTTPS 的 ONTAP Mediator 升级示例（控制台输出）

```

[root@mediator_host~]# ontap-mediator-1.11.0/ontap-mediator-1.11.0

ONTAP Mediator: Self Extracting Installer

+ Extracting the ONTAP Mediator installation/upgrade archive
+ Performing the ONTAP Mediator run-time code signature check
    Using openssl from the path: /usr/bin/openssl configured for
CApath:/etc/pki/tls
+ Unpacking the ONTAP Mediator installer
ONTAP Mediator requires two user accounts. One for the service
(netapp), and one for use by ONTAP to the mediator API

```

```
(mediatoradmin).
```

```
Using default account names: netapp + mediatoradmin
```

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

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

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

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

```
+ Installing required packages.
```

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

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

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

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

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

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

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

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

```
Package libselinux-utils-3.6-3.el9.x86_64 is already installed.
```

```
Package patch-2.7.6-16.el9.x86_64 is already installed.
```

```
Package perl-Data-Dumper-2.174-462.el9.x86_64 is already installed.
```

```
Package perl-ExtUtils-MakeMaker-2:7.60-3.el9.noarch is already  
installed.
```

```
Package bzip2-1.0.8-10.el9_5.x86_64 is already installed.
```

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

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

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

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

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

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

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

```
Dependencies resolved.
```

```

Nothing to do.
Complete!
OS package installations finished
+ Installing ONTAP Mediator. (Log: /root/ontap_mediator.JclgnS/ontap-
mediator-1.11.0/ontap-mediator-1.11.0/install_20260421112205.log)
    This step will take several minutes. Use the log file to view
progress.
    Sudoer config verified
    ONTAP Mediator rsyslog and logging rotation enabled
+ Install successful. (Moving log to
/opt/netapp/lib/ontap_mediator/log/install_20260421112205.log)

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

+ Note: ONTAP Mediator uses a kernel module compiled specifically for
the current
    OS. Using 'yum update' to upgrade the kernel might cause
service interruption.
    For more information, see /opt/netapp/lib/ontap_mediator/README
[root@mediator_host~]#
[root@mediator_host~]# tar -zxvf ontap-mediator-1.12.1.tgz
ontap-mediator-1.12.1/
ontap-mediator-1.12.1/csc-prod-chain-ONTAP-Mediator.pem
ontap-mediator-1.12.1/tsa-prod-ONTAP-Mediator.pem
ontap-mediator-1.12.1/csc-prod-ONTAP-Mediator.pem
ontap-mediator-1.12.1/tsa-prod-chain-ONTAP-Mediator.pem
ontap-mediator-1.12.1/ONTAP-Mediator-production.pub
ontap-mediator-1.12.1/ontap-mediator-1.12.1
ontap-mediator-1.12.1/ontap-mediator-1.12.1.sig
ontap-mediator-1.12.1/ontap-mediator-1.12.1.tsr
ontap-mediator-1.12.1/ontap-mediator-1.12.1.sig.tsr
[root@mediator_host~]#
[root@mediator_host~]# ontap-mediator-1.12.1/ontap-mediator-1.12.1
--https-only

ONTAP Mediator: Self Extracting Installer

+ Extracting the ONTAP Mediator installation/upgrade archive

```

```
+ Performing the ONTAP Mediator run-time code signature check
  Using openssl from the path: /usr/bin/openssl configured for
CApath:/etc/pki/tls
+ Unpacking the ONTAP Mediator installer
+ ONTAP_MEDIATOR_INSTALL_HTTPS_ONLY is set to '1', args are: ''
+ Starting ONTAP Mediator install/update process: previous
installation_mode is ''
+ Selected install package mode: HTTPS_ONLY
ONTAP Mediator will be upgraded from version 1.11.0 to 1.12.1.
```

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

Capturing support_bundle to save pre-upgrade configuration. API
Credentials are required.

Mediator API User Name: mediatoradmin

Password: Running plugins, please wait...

Creating tar archive...

Support bundle has been generated at

/opt/netapp/data/support_bundles/scs001097036-1_1776785423_BASIC.tgz

Testing the DB can be upgraded

Cloning database...

Cloning successful.

Verifying database is up-to-date...

Database out-of-date.Migrations are necessary:

Current revision: 0d5dcf349b27, Head revision: 758e4873c40d

Performing database migrations against clone...

Migrations successful.

Database upgrade operation complete.

+ Adding python312 specific packages

+ Upgrading ONTAP Mediator. (Log: /root/ontap_mediator.cRy0mN/ontap-
mediator-1.12.1/ontap-mediator-1.12.1/upgrade_20260421113003.log)

fapolicyd not installed or no rules.d directory found, skipping
fapolicyd configuration

ONTAP Mediator rsyslog and logging rotation enabled

+ Upgrade successful. (Moving log to

/opt/netapp/lib/ontap_mediator/log/upgrade_20260421113003.log)\n

+ Note: ONTAP Mediator generated a self-signed server certificate for
temporary use on

this host. If the DNS name or IP address for the host is changed,
the certificate

will no longer be valid. The default certificates should be
replaced with secure

trusted certificates signed by a known certificate authority prior

```
to use for production.
```

```
For more information, see /opt/netapp/lib/ontap_mediator/README
```

```
+ Note: ONTAP Mediator uses a kernel module compiled specifically for  
the current
```

```
OS. Using 'yum update' to upgrade the kernel might cause  
service interruption.
```

```
For more information, see /opt/netapp/lib/ontap_mediator/README
```

```
[root@mediator_host ~]#
```

升级主机操作系统并重新安装或升级 **ONTAP Mediator**

要将安装了 SCST 的 ONTAP Mediator 的主机操作系统升级到更高版本，您必须卸载 ONTAP Mediator。操作系统升级后，您可以重新安装当前版本的 ONTAP Mediator 或升级到较新版本。

关于此任务

- 此任务适用于安装了 SCST 的 ONTAP Mediator 版本。如果您安装了仅支持 HTTPS 的 ONTAP Mediator 1.12 或更高版本，则未安装 SCST 软件包，您可以升级主机操作系统而不会影响 ONTAP Mediator。有关操作系统升级说明，请参阅主机操作系统供应商的文档。
- 如果您只需要将主机操作系统内核更新为受支持的版本，而无需升级到新的操作系统版本，请改为参阅 ["维护 ONTAP Mediator 的主机操作系统"](#)。
- 在使用 leapp-upgrade 工具升级 ONTAP Mediator 的主机操作系统之前，["卸载 ONTAP Mediator"](#)。该工具会检查已注册存储库中的新 RPM 版本。
- ONTAP Mediator 安装程序会安装 .rpm 文件，leapp-upgrade 工具会将其包含在搜索中。由于安装程序解压文件而不是从注册存储库下载文件，因此该工具无法找到升级。您必须使用 leapp-upgrade 工具来卸载此软件包。

步骤

1. 备份日志文件：

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

```
[rootmediator-host ~]#
```

2. 使用 leapp-upgrade 工具执行升级:

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

3. 重新安装 ONTAP Mediator:



重新安装 ONTAP Mediator 后立即执行其余步骤，以防止日志文件丢失。

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

ONTAP Mediator: Self Extracting Installer

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

4. 停止 ontap_mediator:

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

5. 替换日志文件:

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

6. 启动 ontap_mediator:

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

7. 将所有 ONTAP 集群重新连接到升级后的 ONTAP 调解器:

MetroCluster 基于 IP

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

siteA::> metrocluster configuration-settings mediator remove
Removing the mediator and disabling Automatic Unplanned Switchover. It
may take a few minutes to complete.
Please enter the username for the mediator: mediatoradmin
Please enter the password for the mediator:
Confirm the mediator password:
Automatic Unplanned Switchover is disabled for all nodes...
Removing mediator mailboxes...
Successfully removed the mediator.

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

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

```
siteA::>
```

SnapMirror 活动同步

对于 SnapMirror 活动同步，您无需重新安装存储在 /opt/netapp 之外的 TLS 证书。备份和还原存储在 /opt/netapp 中的证书。

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

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

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

peer1::> job show -id 39

Job ID Name              Owning
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
```

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

```
peer1::> snapmirror mediator show
```

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

```
peer1::>
```

相关信息

- ["security certificate delete"](#)
- ["security certificate install"](#)
- ["security certificate show"](#)
- ["snapmirror mediator add"](#)
- ["snapmirror mediator remove"](#)
- ["storage iscsi-initiator show"](#)

配置 ONTAP Mediator 属性

您可以配置 ONTAP Mediator 主机端属性，包括与证书相关的设置和配置覆盖。

ONTAP Mediator 可配置设置

ONTAP Mediator 支持多种可配置设置。这些设置的默认值在 `low\_space\_threshold\_mib: 10` 只读文件中提供。该文件位于：

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

放置在 `ontap\_mediator.user\_config.yaml` 中的所有值将覆盖默认值，并在所有 ONTAP Mediator 升级中保留。

修改 `ontap\_mediator.user\_config.yaml` 后，重新启动 ONTAP Mediator：

```
systemctl restart ontap_mediator
```

如果需要，可以修改本节中介绍的 ONTAP Mediator 属性。



ontap_mediator.config.yaml 中的其他默认值不应更改，因为在 ONTAP Mediator 升级期间不会保留修改后的值。

您可以通过将所需变量复制到 `ontap\_mediator.user\_config.yaml` 文件以覆盖默认设置来修改 ONTAP Mediator 属性。

安装第三方 SSL 证书

您可以通过修改 ONTAP Mediator 配置文件，将默认自签名证书替换为第三方 SSL 证书。

要将默认自签名证书替换为第三方 SSL 证书，请修改以下文件中的某些属性：

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

- /opt/netapp/lib/ontap_mediator/uwsgi/ontap_mediator.ini

这些文件中的变量用于控制 ONTAP Mediator 使用的证书文件。

ONTAP Mediator 1.9 及更高版本

`/opt/netapp/lib/ontap\_mediator/ontap\_mediator/server\_config/ontap\_mediator.config.yaml` 文件中包含下表列出的默认变量。

变量	路径
cert_path	/opt/netapp/lib/ontap_mediator/ontap_mediator/server_config/ontap_mediator_server.crt
key_path	/opt/netapp/lib/ontap_mediator/ontap_mediator/server_config/ontap_mediator_server.key
ca_cert_path	/opt/netapp/lib/ontap_mediator/ontap_mediator/server_config/intermediate.crt
ca_key_path	/opt/netapp/lib/ontap_mediator/ontap_mediator/server_config/intermediate.key
ca_serial_path	/opt/netapp/lib/ontap_mediator/ontap_mediator/server_config/intermediate.srl
cert_valid_days	1095
x509_passin_pwd	pass:ontap

- cert_valid_days 用于设置客户端证书的到期时间。最大值为三年（1095 天）。
- x509_passin_pwd 是已签名客户端证书的密码短语。

下表列出的默认变量包含在 /opt/netapp/lib/ontap_mediator/uwsgi/ontap_mediator.ini 文件中。

变量	路径
mediator_cert	/opt/netapp/lib/ontap_mediator/ontap_mediator/server_config/ontap_mediator_server.crt
mediator_key	/opt/netapp/lib/ontap_mediator/ontap_mediator/server_config/ontap_mediator_server.key
ca_cert_path	/opt/netapp/lib/ontap_mediator/ontap_mediator/server_config/intermediate.crt

ONTAP Mediator 1.8 及更早版本

``/opt/netapp/lib/ontap_mediator/ontap_mediator/server_config/ontap_mediator.config.yaml`` 文件中包含下表列出的默认变量。

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

- cert_valid_days 用于设置客户端证书的到期时间。最大值为三年（1095 天）。
- x509_passin_pwd 是已签名客户端证书的密码短语。

下表列出的默认变量包含在 `/opt/netapp/lib/ontap_mediator/uwsgi/ontap_mediator.ini` 文件中。

变量	路径
mediator_cert	/opt/netapp/lib/ontap_mediator/ontap_mediator/server_config/ontap_mediator_server.crt
mediator_key	/opt/netapp/lib/ontap_mediator/ontap_mediator/server_config/ontap_mediator_server.key
ca_cert_path	/opt/netapp/lib/ontap_mediator/ontap_mediator/server_config/ca.crt

如果修改这些属性，请重新启动 ONTAP Mediator 以应用更改。有关如何使用第三方证书替换默认证书的详细说明，请参阅 ["用受信任的第三方证书替换自签名证书"](#)。

配置密码攻击防护

以下设置可提供针对暴力破解密码猜测攻击的保护。

要启用该功能，请设置 `window\_seconds` 和 `retry\_limit` 的值。

示例：

- 提供 5 分钟的猜测窗口，然后将计数重置为零失败：

```
authentication_lock_window_seconds: 300
```

- 如果在窗口期内发生五次失败，请锁定帐户：

```
authentication_retry_limit: 5
```

- 通过设置在拒绝每次尝试之前发生的延迟来降低暴力密码猜测攻击的影响，从而减缓攻击速度。

```
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
```

配置密码复杂性规则

以下字段控制 ONTAP Mediator API 用户帐户的密码复杂性规则。

```
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)
```

配置 ONTAP Mediator 磁盘和日志空间设置，包括可用空间阈值和保留的日志空间。

配置可用空间阈值

有一些设置可以控制 `/opt/netapp/lib/ontap\_mediator` 磁盘上所需的可用空间。

如果空间低于设置的阈值，则服务将发出警告事件。

```
low_space_threshold_mib: 10
```

配置预留日志空间

该 `RESERVE\_LOG\_SPACE` 功能由特定设置控制。默认情况下，ONTAP Mediator 安装会为日志创建一个单独的磁盘空间。安装程序会创建一个总计 700MB 磁盘空间的固定大小新文件，专门用于 ONTAP Mediator 日志记录。

要禁用此功能并使用默认磁盘空间，请执行以下步骤：

1. 在以下文件中，将 RESERVE_LOG_SPACE 的值从 1 更改为 0：

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

- a. 运行以下命令：

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

```
RESERVE_LOG_SPACE=0
```

- b. 重新启动 ONTAP Mediator：

```
systemctl restart ontap_mediator
```

要重新启用此功能，请将值从 0 更改为 1，然后重新启动 Mediator。



在磁盘空间之间切换不会清除现有日志。切换并重新启动 Mediator 后，将备份所有以前的日志，然后将其移动到当前磁盘空间。

更改 ONTAP Mediator 用户名和密码

从安装了 ONTAP Mediator 的 Linux 主机更改 ONTAP Mediator 用户名或密码。

更改用户名

可以使用以下步骤更改用户名。

关于此任务

在已安装 ONTAP Mediator 的 Linux 主机上执行此任务。

如果无法访问此命令，则可能需要使用完整路径运行该命令，如以下示例中所示：

/usr/local/bin/mediator_username

步骤

通过选择以下选项之一更改用户名：

- 选项 **(a)**：运行命令 `mediator\_change\_user` 并响应提示，如下例所示：

```
[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 ~]#
```

- 选项 **(b)**：运行以下命令：

```
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 ~]#
```

更改密码

您可以使用以下步骤更改密码。

关于此任务

在已安装 ONTAP Mediator 的 Linux 主机上执行此任务。

如果无法访问此命令，则可能需要使用完整路径运行该命令，如以下示例中所示：

/usr/local/bin/mediator_change_password

步骤

通过选择以下选项之一更改密码：

- 选项 **(a)**：运行 `mediator\_change\_password` 命令并响应提示，如以下示例所示：

```
[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 ~]#
```

- 选项 (b) : 运行以下命令:

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

示例显示密码从“mediator1”更改为“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 ~]#
```

启动和停止 ONTAP Mediator

需要维护主机或重新启用服务时，停止或重新启动 ONTAP Mediator 和 SCST。

停止 ONTAP Mediator

要停止 ONTAP Mediator，请执行以下步骤：

步骤

1. 停止 ONTAP Mediator:

```
systemctl stop ontap_mediator
```

2. 停止 SCST:

```
systemctl stop mediator-scst
```

3. 禁用 ONTAP 调解器和 SCST:

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

启动 ONTAP Mediator

要重新启用 ONTAP Mediator，请执行以下步骤：

步骤

1. 启用 ONTAP 调解器和 SCST:

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

2. 启动 SCST:

```
systemctl start mediator-scst
```

3. 启动 ONTAP Mediator:

```
systemctl start ontap_mediator
```

验证 ONTAP Mediator 运行状况

验证 ONTAP Mediator 是否正常运行。

验证服务状态

步骤

1. 查看 ONTAP Mediator 的状态:

- a. 运行以下命令:

```
systemctl status ontap_mediator
```

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

ontap_mediator.service - ONTAP Mediator
Loaded: loaded (/etc/systemd/system/ontap_mediator.service; enabled;
vendor preset: disabled)
Active: active (running) since Mon 2022-04-18 10:41:49 EDT; 1 weeks 0
days ago
Process: 286710 ExecStop=/bin/kill -s INT $MAINPID (code=exited,
status=0/SUCCESS)
Main PID: 286712 (uwsgi)
Status: "uWSGI is ready"
```

```
Tasks: 3 (limit: 49473)
Memory: 139.2M
CGroup: /system.slice/ontap_mediator.service
├─286712 /opt/netapp/lib/ontap_mediator/pyenv/bin/uwsgi --ini
/opt/netapp/lib/ontap_mediator/uwsgi/ontap_mediator.ini
├─286716 /opt/netapp/lib/ontap_mediator/pyenv/bin/uwsgi --ini
/opt/netapp/lib/ontap_mediator/uwsgi/ontap_mediator.ini
└─286717 /opt/netapp/lib/ontap_mediator/pyenv/bin/uwsgi --ini
/opt/netapp/lib/ontap_mediator/uwsgi/ontap_mediator.ini

[root@mediator_host ~]#
```

b. 查看 SCST 服务的状态:



如果您安装了仅支持 HTTPS 的 ONTAP Mediator 1.12 或更高版本, 则未安装 SCST 包, 您可以跳过此任务。

```
systemctl status mediator-scst
```

+

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

[root@mediator_host ~]#
```

确认侦听端口

1. 确认 ONTAP Mediator 使用的端口:

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

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

```

tcp      0      0 0.0.0.0:31784      0.0.0.0:*          LISTEN
tcp      0      0 0.0.0.0:3260       0.0.0.0:*          LISTEN
tcp6     0      0 :::3260            :::*                LISTEN

```

管理证书

重新生成 **ONTAP Mediator** 主机的临时自签名证书

当主机名或 IP 地址更改使现有证书无效时，为当前主机重新生成临时自签名证书。



此过程仅在运行 ONTAP Mediator 1.7 或更高版本的系统上受支持。

关于此任务

- 在已安装 ONTAP Mediator 的 Linux 主机上执行此任务。
- 只有在安装 ONTAP Mediator 后，由于主机的主机名或 IP 地址发生更改而导致生成的自签名证书过时，您才能执行此任务。
- 临时自签名证书被受信任的第三方证书替换后，请不要使用此任务重新生成证书。缺少自签名证书将导致此过程失败。

步骤

要为当前主机重新生成新的临时自签名证书，请执行以下步骤：

1. 重新启动 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

```

将 ONTAP Mediator 自签名证书替换为受信任的第三方证书

如果环境支持，请将自签名证书替换为由证书颁发机构颁发的可信证书。

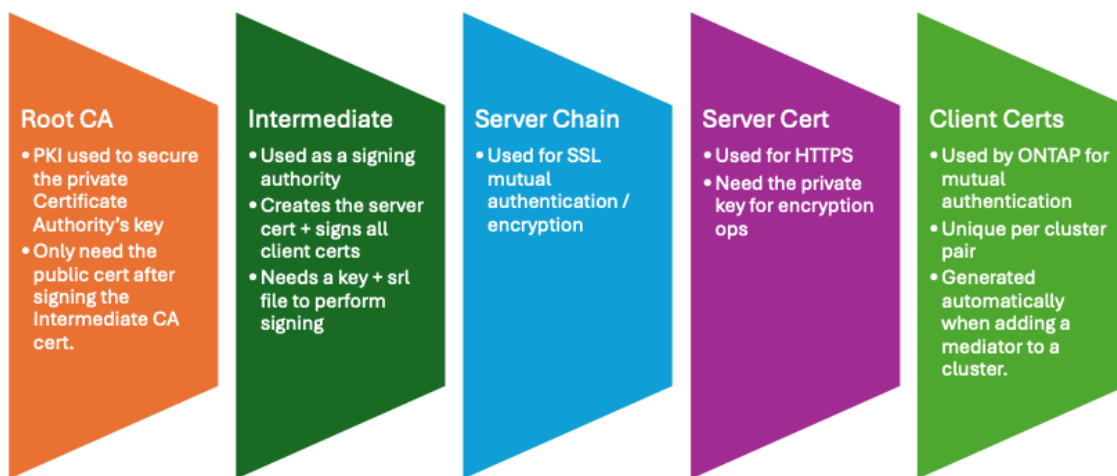


- 仅从 ONTAP 9.16.1 开始以及在某些较早的 ONTAP 补丁版本中支持第三方证书。请参阅["NetApp Bugs Online Bug ID CONTAP-243278"](#)。
- 仅在运行 ONTAP Mediator 1.7 或更高版本的系统上支持第三方证书。

关于此任务

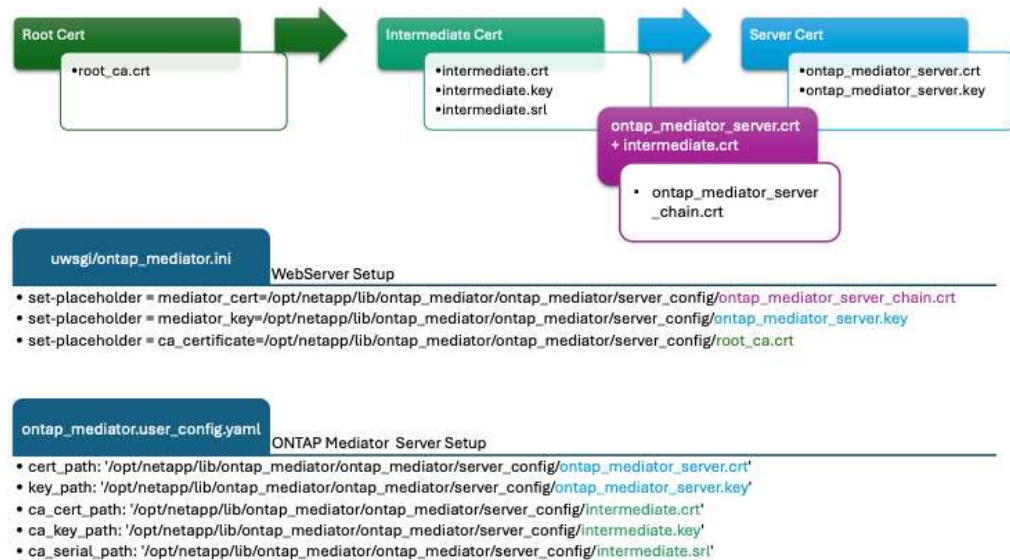
- 在已安装 ONTAP Mediator 的 Linux 主机上执行此任务。
- 如果需要将生成的自签名证书替换为从受信任的从属证书颁发机构（CA）获得的证书，则可以执行此任务。为了实现这一点，您应该有权访问受信任的公钥基础设施（PKI）机构。
- 下图显示了每个 ONTAP Mediator 证书的用途。

ONTAP Mediator Certificate Purposes



- 下图显示了 Web 服务器设置和 ONTAP Mediator 设置的配置。

ONTAP Mediator Certificates



步骤 1: 从颁发 CA 证书的第三方获取证书

您可以通过以下步骤从 PKI 机构获取证书。

以下示例演示了将自签名证书参与者替换为位于 `/opt/netapp/lib/ontap\_mediator/ontap\_mediator/server\_config/` 的第三方证书参与者。



本示例说明了 ONTAP Mediator 所需证书的必要标准。您可以通过与此过程不同的方式从 PKI 机构获取证书。请根据您的业务需求调整此过程。

ONTAP Mediator 1.9 及更高版本

1. 创建一个私钥 `intermediate.key` 和一个配置文件 `openssl_ca.cnf`，供 PKI 机构用于生成证书。
 - a. 生成私钥 `intermediate.key`:
 示例

```
openssl genrsa -aes256 -out intermediate.key 4096
```
 - b. 配置文件 `openssl_ca.cnf` (位于 `/opt/netapp/lib/ontap_mediator/ontap_mediator/server_config/openssl_ca.cnf`) 定义了生成的证书必须具有的属性。
2. 使用私钥和配置文件创建证书签名请求 `intermediate.csr`:
 示例:

```
openssl req -key <private_key_name>.key -new -out
```

```
<certificate_csr_name>.csr -config <config_file_name>.cnf
```

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

3. 将证书签名请求 `intermediate.csr` 发送给 PKI 机构进行签名。

PKI 机构验证请求并对 `.csr` 进行签名，从而生成证书 `intermediate.crt`。此外，您还需要从 PKI 机构获取对 `intermediate.crt` 证书进行签名的 `root\_ca.crt` 证书。



对于使用配置了 HTTPS 的 ONTAP Mediator 的 SnapMirror 活动同步集群和 MetroCluster IP 配置，您必须将 `intermediate.crt` 和 `root\_ca.crt` 证书添加到 ONTAP 集群。请参阅["为 SnapMirror 活动同步配置 ONTAP 调解器和集群"](#)和["为 MetroCluster IP 配置配置 ONTAP Mediator"](#)。

ONTAP Mediator 1.8 及更早版本

1. 创建私钥 `ca.key` 和配置文件 `openssl\_ca.cnf`，供 PKI 机构用来生成证书。

- a. 生成私钥 `ca.key`:

示例:

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

- b. 配置文件 `openssl\_ca.cnf` (位于

/opt/netapp/lib/ontap_mediator/ontap_mediator/server_config/openssl_ca.cnf) 定义了生成的证书必须具有的属性。

2. 使用私钥和配置文件创建证书签名请求 `ca.csr`:

示例:

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

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

3. 将证书签名请求 `ca.csr` 发送给 PKI 机构进行签名。

PKI 机构验证请求并对 `ca.csr` 进行签名，从而生成证书 `ca.crt`。此外，您还需要从 PKI 机构获取对 `ca.crt` 证书进行签名的 `root\_ca.crt` 证书。



对于 SnapMirror 活动同步集群，必须将 `ca.crt` 和 `root\_ca.crt` 证书添加到 ONTAP 集群。请参阅[“为 SnapMirror 活动同步配置 ONTAP 调解器和集群”](#)。

步骤 2：通过使用第三方 CA 证书签名来生成服务器证书

ONTAP Mediator 1.9 及更高版本

服务器证书必须由私钥 `intermediate.key` 和第三方证书 `intermediate.crt` 签名。此外，配置文件 `/opt/netapp/lib/ontap\_mediator/ontap\_mediator/server\_config/openssl\_server.cnf` 包含某些属性，这些属性指定由 OpenSSL 颁发的服务器证书所需的属性。

以下命令可以生成服务器证书。

步骤

1. 要生成服务器证书签名请求（CSR），请从
`/opt/netapp/lib/ontap\_mediator/ontap\_mediator/server\_config` 文件夹中运行以下命令：

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

2. 要从 CSR 生成服务器证书，请从
`/opt/netapp/lib/ontap\_mediator/ontap\_mediator/server\_config` 文件夹中运行以下命令：



这些文件是从 PKI 机构获得的。如果使用不同的证书名称，请将 `intermediate.crt` 和 `intermediate.key` 替换为相关文件名。

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

- 该 `-CAcreateserial` 选项用于生成 `intermediate.srl` 文件。

ONTAP Mediator 1.8 及更早版本

服务器证书必须由私钥 `ca.key` 和第三方证书 `ca.crt` 签名。此外，配置文件 `/opt/netapp/lib/ontap\_mediator/ontap\_mediator/server\_config/openssl\_server.cnf` 包含某些属性，这些属性指定由 OpenSSL 颁发的服务器证书所需的属性。

以下命令可以生成服务器证书。

步骤

1. 要生成服务器证书签名请求（CSR），请从

/opt/netapp/lib/ontap_mediator/ontap_mediator/server_config 文件夹中运行以下命令：

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

2. 要从 CSR 生成服务器证书，请从

/opt/netapp/lib/ontap_mediator/ontap_mediator/server_config 文件夹中运行以下命令：



这些文件是从 PKI 机构获得的。如果使用不同的证书名称，请将 `ca.crt` 和 `ca.key` 替换为相关文件名。

```
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
```

◦ 该 `-CAcreateserial` 选项用于生成 `ca.srl` 文件。

步骤 3：在 ONTAP Mediator 配置中替换新的第三方 CA 证书和服务器证书

ONTAP Mediator 1.10 及更高版本

证书配置在位于

`/opt/netapp/lib/ontap\_mediator/ontap\_mediator/server\_config/ontap\_mediator.config.yaml` 的配置文件中提供给 ONTAP Mediator。该文件包含以下属性：

```
cert_path:  
'/opt/netapp/lib/ontap_mediator/ontap_mediator/server_config/ontap_medi  
ator_server.crt'  
key_path:  
'/opt/netapp/lib/ontap_mediator/ontap_mediator/server_config/ontap_medi  
ator_server.key'  
ca_cert_path:  
'/opt/netapp/lib/ontap_mediator/ontap_mediator/server_config/intermedia  
te.crt'  
ca_key_path:  
'/opt/netapp/lib/ontap_mediator/ontap_mediator/server_config/intermedia  
te.key'  
ca_serial_path:  
'/opt/netapp/lib/ontap_mediator/ontap_mediator/server_config/intermedia  
te.srl'
```

- cert_path 和 key_path 是服务器证书变量。
- ca_cert_path、`ca\_key\_path` 和 `ca\_serial\_path` 是 CA 证书变量。

步骤

1. 将所有 `intermediate.*` 文件替换为第三方证书。
2. 从 `intermediate.crt` 和 `ontap_mediator_server.crt` 证书创建证书链：

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

3. 更新 `/opt/netapp/lib/ontap_mediator/unicorn/config.json` 文件。

更新 `ssl_keyfile`、`ssl_certfile` 和 `ssl_ca_certs` 的值：

```
ssl_keyfile:
/opt/netapp/lib/ontap_mediator/ontap_mediator/server_config/ontap_mediator_
server.key
```

```
ssl_certfile:
/opt/netapp/lib/ontap_mediator/ontap_mediator/server_config/ontap_mediator_
server_chain.crt
```

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

- `ssl_keyfile` 值是 `ontap_mediator_server.crt` 文件中的键路径，即 `ontap_mediator_server.key`。
- `ssl_certfile` 值是 `ontap_mediator_server_chain.crt` 文件的路径。
- `ssl_ca_certs` 值是 `root_ca.crt` 文件的路径。

4. 验证新生成的证书的以下属性设置是否正确：

- Linux 组所有者： `netapp:netapp`
- Linux 权限： `600`

5. 重新启动 ONTAP Mediator：

```
systemctl restart ontap_mediator
```

ONTAP 调解器 1.9.1 和 1.9

证书配置在位于

`/opt/netapp/lib/ontap_mediator/ontap_mediator/server_config/ontap_mediator.config.yaml` 的配置文件中提供给 ONTAP Mediator。该文件包含以下属性：

```
cert_path:
'/opt/netapp/lib/ontap_mediator/ontap_mediator/server_config/ontap_medi
ator_server.crt'
key_path:
'/opt/netapp/lib/ontap_mediator/ontap_mediator/server_config/ontap_medi
ator_server.key'
ca_cert_path:
'/opt/netapp/lib/ontap_mediator/ontap_mediator/server_config/intermedia
```

```
te.crt'  
ca_key_path:  
'/opt/netapp/lib/ontap_mediator/ontap_mediator/server_config/intermedia  
te.key'  
ca_serial_path:  
'/opt/netapp/lib/ontap_mediator/ontap_mediator/server_config/intermedia  
te.srl'
```

- cert_path 和 key_path 是服务器证书变量。
- ca_cert_path、`ca\_key\_path`和`ca\_serial\_path`是 CA 证书变量。

步骤

1. 将所有 intermediate.* 文件替换为第三方证书。
2. 从`intermediate.crt`和`ontap\_mediator\_server.crt`证书创建证书链：

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

3. 更新`/opt/netapp/lib/ontap\_mediator/uwsgi/ontap\_mediator.ini`文件。

更新 mediator_cert、`mediator\_key`和`ca\_certificate`的值：

```
set-placeholder = mediator_cert =  
/opt/netapp/lib/ontap_mediator/ontap_mediator/server_config/ontap_mediator_  
server_chain.crt
```

```
set-placeholder = mediator_key =  
/opt/netapp/lib/ontap_mediator/ontap_mediator/server_config/ontap_mediator_  
server.key
```

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

- mediator_cert 值是 ontap_mediator_server_chain.crt 文件的路径。
- mediator_key value 是 ontap_mediator_server.crt`文件中的密钥路径，即`ontap_mediator_server.key`。
- ca_certificate 值是`root\_ca.crt`文件的路径。

4. 验证新生成的证书的以下属性设置是否正确：

- Linux 组所有者：netapp:netapp
- Linux 权限：600

5. 重新启动 ONTAP Mediator：

```
systemctl restart ontap_mediator
```

ONTAP Mediator 1.8 及更早版本

证书配置在位于

`/opt/netapp/lib/ontap_mediator/ontap_mediator/server_config/ontap_mediator.config.yaml` 的配置文件中提供给 ONTAP Mediator。该文件包含以下属性：

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

- `cert_path` 和 `key_path` 是服务器证书变量。
- `ca_cert_path`、`ca_key_path` 和 `ca_serial_path` 是 CA 证书变量。

步骤

1. 将所有 `ca.*` 文件替换为第三方证书。
2. 从 `ca.crt` 和 `ontap_mediator_server.crt` 证书创建证书链：

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

3. 更新 `/opt/netapp/lib/ontap_mediator/uwsgi/ontap_mediator.ini` 文件。

更新 `mediator_cert`、`mediator_key` 和 `ca_certificate` 的值：

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

```
set-placeholder = mediator_key =
/opt/netapp/lib/ontap_mediator/ontap_mediator/server_config/ontap_mediator_server.key
```

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

- `mediator_cert` 值是 `ontap_mediator_server_chain.crt` 文件的路径。
- `mediator_key` value 是 `ontap_mediator_server.crt` 文件中的密钥路径，即 `ontap_mediator_server.key`。
- `ca_certificate` 值是 `root_ca.crt` 文件的路径。

4. 验证新生成的证书的以下属性设置是否正确：

- Linux 组所有者： netapp:netapp
- Linux 权限： 600

5. 重新启动 ONTAP Mediator：

```
systemctl restart ontap_mediator
```

步骤 4：（可选）为第三方证书使用其他路径或名称

ONTAP Mediator 1.10 及更高版本

您可以使用名称不同于 `intermediate.\*` 的第三方证书，或者将第三方证书存储在其他位置。

步骤

1. 配置

/opt/netapp/lib/ontap_mediator/ontap_mediator/server_config/ontap_mediator.user_config.yaml 文件以覆盖 ontap_mediator.config.yaml 文件中的默认变量值。

如果您从 PKI 机构获取了 intermediate.crt，并将其私钥 intermediate.key 存储在位置 /opt/netapp/lib/ontap_mediator/ontap_mediator/server_config，则 `ontap\_mediator.user\_config.yaml` 文件应如下所示：



如果您使用 `intermediate.crt` 签署了 `ontap\_mediator\_server.crt` 证书，则会生成 `intermediate.srl` 文件。有关详细信息，请参见 [步骤 2：通过使用第三方 CA 证书签名来生成服务器证书](#)。

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

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

```

'/opt/netapp/lib/ontap_mediator/ontap_mediator/server_config/ontap_mediator_server.crt'
key_path:
'/opt/netapp/lib/ontap_mediator/ontap_mediator/server_config/ontap_mediator_server.key'
ca_cert_path:
'/opt/netapp/lib/ontap_mediator/ontap_mediator/server_config/intermediate.crt'
ca_key_path:
'/opt/netapp/lib/ontap_mediator/ontap_mediator/server_config/intermediate.key'
ca_serial_path:
'/opt/netapp/lib/ontap_mediator/ontap_mediator/server_config/intermediate.srl'

```

- a. 如果您使用的证书结构中， root_ca.crt 证书提供了一个 intermediate.crt 证书来签署 ontap_mediator_server.crt 证书，请从 intermediate.crt 和 ontap_mediator_server.crt 证书创建证书链：



您应该在此过程的早期从 PKI 机构获取了 `intermediate.crt` 和 `ontap\_mediator\_server.crt` 证书。

```

cat ontap_mediator_server.crt intermediate.crt >
ontap_mediator_server_chain.crt

```

- b. 更新 `/opt/netapp/lib/ontap\_mediator/unicorn/config.json` 文件。

更新 ssl_keyfile、`ssl\_certfile` 和 `ssl\_ca\_certs` 的值：

```

ssl_keyfile:
/opt/netapp/lib/ontap_mediator/ontap_mediator/server_config/ontap_mediator_server.key

```

```

ssl_certfile:
/opt/netapp/lib/ontap_mediator/ontap_mediator/server_config/ontap_mediator_server_chain.crt

```

```

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

```

- ssl_keyfile 值是 ontap_mediator_server.crt 文件中的键路径，即 ontap_mediator_server.key。
- ssl_certfile 值是 ontap_mediator_server_chain.crt 文件的路径。
- ssl_ca_certs 值是 root_ca.crt 文件的路径。



对于使用配置了 HTTPS 的 ONTAP Mediator 的 SnapMirror 活动同步集群和 MetroCluster IP 配置，您必须将 `intermediate.crt` 和 `root\_ca.crt` 证书添加到 ONTAP 集群。请参阅[为 SnapMirror 活动同步配置 ONTAP 调解器和集群](#)和[为](#)

c. 验证新生成的证书的以下属性设置是否正确：

- Linux 组所有者： netapp:netapp
- Linux 权限： 600

2. 在配置文件中更新证书时重新启动 ONTAP Mediator：

```
systemctl restart ontap_mediator
```

ONTAP 调解器 1.9.1 和 1.9

您可以使用名称不同于 `intermediate.\*` 的第三方证书，或者将第三方证书存储在其他位置。

步骤

1. 配置

/opt/netapp/lib/ontap_mediator/ontap_mediator/server_config/ontap_mediator.user_config.yaml 文件以覆盖 ontap_mediator.config.yaml 文件中的默认变量值。

如果您从 PKI 机构获取了 intermediate.crt，并将其私钥 intermediate.key 存储在位置 `/opt/netapp/lib/ontap\_mediator/ontap\_mediator/server\_config`，则 `ontap\_mediator.user\_config.yaml` 文件应如下所示：



如果您使用 `intermediate.crt` 签署了 `ontap\_mediator\_server.crt` 证书，则会生成 `intermediate.srl` 文件。有关详细信息，请参见 [步骤 2：通过使用第三方 CA 证书签名来生成服务器证书](#)。

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

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

```
key_path:
'/opt/netapp/lib/ontap_mediator/ontap_mediator/server_config/ontap_m
ediator_server.key'
ca_cert_path:
'/opt/netapp/lib/ontap_mediator/ontap_mediator/server_config/interme
diate.crt'
ca_key_path:
'/opt/netapp/lib/ontap_mediator/ontap_mediator/server_config/interme
diate.key'
ca_serial_path:
'/opt/netapp/lib/ontap_mediator/ontap_mediator/server_config/interme
diate.srl'
```

- a. 如果您使用的证书结构中，`root_ca.crt` 证书提供了一个 `intermediate.crt` 证书来签署 `ontap_mediator_server.crt` 证书，请从 `intermediate.crt` 和 `ontap_mediator_server.crt` 证书创建证书链：



您应该在此过程的早期从 PKI 机构获取了 `'intermediate.crt'` 和 `'ontap_mediator_server.crt'` 证书。

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

- b. 更新 `'/opt/netapp/lib/ontap_mediator/uwsgi/ontap_mediator.ini'` 文件。

更新 `mediator_cert`、`'mediator_key'` 和 `'ca_certificate'` 的值：

```
set-placeholder = mediator_cert =
/opt/netapp/lib/ontap_mediator/ontap_mediator/server_config/ontap_mediato
r_server_chain.crt
```

```
set-placeholder = mediator_key =
/opt/netapp/lib/ontap_mediator/ontap_mediator/server_config/ontap_mediato
r_server.key
```

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

- `mediator_cert` 值是 `ontap_mediator_server_chain.crt` 文件的路径。
- `mediator_key` 值是 `ontap_mediator_server.crt` 文件中的键路径，即 `'ontap_mediator_server.key'`。
- `ca_certificate` 值是 `'root_ca.crt'` 文件的路径。



对于 SnapMirror 业务连续性 (SM-BC) 集群，必须将 `intermediate.crt` 和 `root_ca.crt` 证书添加到 ONTAP 集群。请参阅 ["为 SnapMirror 活动同步配置 ONTAP 调解器和集群"](#)。

c. 验证新生成的证书的以下属性设置是否正确：

- Linux 组所有者： netapp:netapp
- Linux 权限： 600

2. 在配置文件中更新证书时重新启动 ONTAP Mediator：

```
systemctl restart ontap_mediator
```

ONTAP Mediator 1.8 及更早版本

您可以使用名称不同于 `ca.\*` 的第三方证书，也可以将第三方证书存储在其他位置。

步骤

1. 配置

/opt/netapp/lib/ontap_mediator/ontap_mediator/server_config/ontap_mediator.user_config.yaml 文件以覆盖 ontap_mediator.config.yaml 文件中的默认变量值。

如果您从 PKI 机构获取了 ca.crt，并将其私钥 ca.key 存储在位置

`/opt/netapp/lib/ontap\_mediator/ontap\_mediator/server\_config`，则 `ontap\_mediator.user\_config.yaml` 文件应如下例所示：



如果您使用 `ca.crt` 对 `ontap\_mediator\_server.crt` 证书进行了签名，则会生成 `ca.srl` 文件。有关详细信息，请参见 [步骤 2：通过使用第三方 CA 证书签名来生成服务器证书](#)。

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

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

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

- a. 如果您使用的证书结构中，`root_ca.crt` 证书提供了一个 `ca.crt` 证书来签署 `ontap_mediator_server.crt` 证书，请从 `ca.crt` 和 `ontap_mediator_server.crt` 证书创建证书链：



您应该在此过程的早期从 PKI 机构获得了 `'ca.crt'` 和 `'ontap_mediator_server.crt'` 证书。

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

- b. 更新 `'/opt/netapp/lib/ontap_mediator/uwsgi/ontap_mediator.ini'` 文件。

更新 `mediator_cert`、`'mediator_key'` 和 `'ca_certificate'` 的值：

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

```
set-placeholder = mediator_key =
/opt/netapp/lib/ontap_mediator/ontap_mediator/server_config/ontap_mediator_server.key
```

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

- `mediator_cert` 值是 `ontap_mediator_server_chain.crt` 文件的路径。
- `mediator_key` 值是 `ontap_mediator_server.crt` 文件中的键路径，即 `'ontap_mediator_server.key'`。
- `ca_certificate` 值是 `'root_ca.crt'` 文件的路径。



对于 SnapMirror 业务连续性 (SM-BC) 集群，必须将 `'ca.crt'` 和 `'root_ca.crt'` 证书添加到 ONTAP 集群。请参阅["为 SnapMirror 活动同步配置 ONTAP 调解器和集群"](#)。

- c. 验证新生成的证书的以下属性设置是否正确：

- Linux 组所有者： `netapp:netapp`
- Linux 权限： `600`

2. 在配置文件中更新证书时重新启动 ONTAP Mediator：

```
systemctl restart ontap_mediator
```

对 **ONTAP Mediator** 证书相关问题进行故障诊断

在对证书问题进行故障排除时，使用这些检查来验证证书属性。

验证证书过期时间

使用以下命令来标识证书有效期范围。

ONTAP Mediator 1.9 及更高版本

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

ONTAP Mediator 1.8 及更早版本

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

验证 **CA** 认证中的 **X509v3** 扩展

使用以下命令验证 CA 认证中的 X509v3 扩展。

ONTAP Mediator 1.9 及更高版本

在 `openssl_ca.cnf` 中的 ``v3_ca`` 内定义的属性将在 ``intermediate.crt`` 中显示为 ``X509v3 extensions``。

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

[root@mediator_host server_config]# cat openssl_ca.cnf
...
```

```
[ v3_ca ]
subjectKeyIdentifier = hash
authorityKeyIdentifier = keyid:always,issuer
basicConstraints = critical, CA:true
keyUsage = critical, cRLSign, digitalSignature, keyCertSign

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

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

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

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

ONTAP Mediator 1.8 及更早版本

在 openssl_ca.cnf 中的 `v3\_ca` 内定义的属性在 `ca.crt` 中显示为 `X509v3 extensions`。

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

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

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

```

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

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

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

```

验证服务器证书和主题替代名称中的 **X509v3** 扩展

`openssl\_server.cnf` 配置文件中定义的 `v3\_req` 属性在证书中显示为 `X509v3 extensions`。

在以下示例中，您可以通过在安装了 ONTAP Mediator 的 Linux VM 上运行命令 `hostname -A` 和 `hostname -I` 来获取 `alt\_names` 部分中的变量。

请与网络管理员联系以获取变量的正确值。

ONTAP Mediator 1.9 及更高版本

```

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

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

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

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

```

```

...

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

```

ONTAP Mediator 1.8 及更早版本

```

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

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

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

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

X509v3 extensions:
  X509v3 Basic Constraints:
    CA:FALSE
  X509v3 Extended Key Usage:
    TLS Web Server Authentication
  X509v3 Key Usage:
    Key Encipherment, Data Encipherment
  X509v3 Subject Alternative Name:

```

```
DNS:abc.company.com, DNS:abc-v6.company.com, IP
Address:1.2.3.4, IP Address:abcd:abcd:abcd:abcd:abcd:abcd
```

验证私钥是否与证书匹配

您可以验证特定私钥是否与证书匹配。

分别在密钥和证书上使用以下 OpenSSL 命令。

ONTAP Mediator 1.9 及更高版本

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

ONTAP Mediator 1.8 及更早版本

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

如果两者的 `-modulus` 属性匹配，则表明私钥和证书对是兼容的，并且可以协同工作。

确认从特定 CA 证书创建了服务器证书

您可以使用以下命令验证服务器证书是否是从特定 CA 证书创建的。

ONTAP Mediator 1.9 及更高版本

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

ONTAP Mediator 1.8 及更早版本

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

如果正在使用在线证书状态协议（OCSP）验证，请使用命令 ["openssl-verify"](#)。

维护 ONTAP Mediator 的主机操作系统

确保 ONTAP Mediator 的主机操作系统定期维护，以实现最佳性能。

重新启动主机

仅在集群运行状况良好时重新启动主机。ONTAP Mediator 脱机时，集群无法响应故障。在重新启动之前设置维护时间。

ONTAP Mediator 在重新启动期间自动恢复，并重新进入先前配置的与 ONTAP 集群的关系。

主机包更新

更新除内核之外的任何库或 yum 软件包。如果需要使更改生效，请重新启动主机。在重新启动主机之前，请安排一个服务窗口。

如果安装 `yum-utils` 软件包，请使用 `needs-restarting` 命令检测是否有任何软件包更改需要重新启动。

更新 ONTAP Mediator 依赖项后重新启动，因为更改不会立即生效。

升级主机 OS 内核

必须为您正在使用的内核编译 SCST。要更新操作系统，您需要安排维护时间。

开始之前

使用 ONTAP Mediator 安装 SCST 包时，适用以下要求。

- 您可以更新除内核之外的所有库包，但可能需要重新启动才能在 ONTAP Mediator 中应用更改。如果需要重新启动，请安排停机时间。
- 您应该使操作系统内核保持最新。请将内核核心升级到 ["ONTAP Mediator 版本矩阵"](#) 中列出的受支持版本。您必须重新启动系统，因此请为停机规划维护时段。
 - 请在重新启动之前卸载 SCST 内核模块，然后重新安装。
 - 在开始内核操作系统升级之前，请准备好要重新安装的受支持的 SCST 版本。



如果您安装了仅支持 HTTPS 的 ONTAP Mediator 1.12 或更高版本，则不会安装 SCST 包，并且可以升级主机操作系统内核，而不会影响 ONTAP Mediator。有关升级说明，请参阅主机操作系统文档。

步骤

按照以下步骤升级主机操作系统内核。



在升级内核之前，请检查操作系统和 ONTAP Mediator 版本是否兼容。有关受支持的版本，请参见["操作系统支持矩阵"](#)。

1. 停止 ONTAP Mediator。
2. 卸载 SCST 包，请参见[\[执行主机维护\]](#)。（SCST 不提供升级机制。）
3. 升级操作系统，然后重新启动。
4. 重新安装 SCST 软件包。
5. 重新启用 ONTAP Mediator。

执行主机维护

升级 VM 内核可能会导致与 SCST 模块的兼容性问题。手动卸载并重新安装 SCST。



如果您安装了仅支持 HTTPS 的 ONTAP Mediator 1.12 或更高版本，则未安装 SCST 包，您可以跳过此任务。

步骤 1: 卸载 SCST

要卸载 SCST，请使用适用于您的 ONTAP Mediator 版本的 tar 捆绑包。

步骤

1. 下载相应的 SCST 捆绑包（如下表所示）并进行解压缩。

对于此版本.....	使用此 tar 捆绑包.....
ONTAP Mediator 1.12.1	scst-3.10.tar.gz
ONTAP Mediator 1.12	scst-3.10.tar.gz
ONTAP Mediator 1.11	scst-3.9.tar.gz
ONTAP Mediator 1.10	scst-3.9.tar.gz
ONTAP Mediator 1.9.1	scst-3.8.0.tar.bz2
ONTAP Mediator 1.9	scst-3.8.0.tar.bz2
ONTAP Mediator 1.8	scst-3.8.0.tar.bz2
ONTAP Mediator 1.7	scst-3.7.0.tar.bz2
ONTAP Mediator 1.6	scst-3.7.0.tar.bz2
ONTAP Mediator 1.5	scst-3.6.0.tar.bz2
ONTAP Mediator 1.4	scst-3.6.0.tar.bz2
ONTAP Mediator 1.3	scst-3.5.0.tar.bz2
ONTAP Mediator 1.1	scst-3.4.0.tar.bz2
ONTAP Mediator 1.0	scst-3.3.0.tar.bz2

- a. 从["SCST sourceforge 下载"](#)访问开源软件包。
 - b. 选择*下载已发布的版本*。
 - c. 将捆绑包提取到您的 VM。
2. 在 `scst` 目录中运行以下卸载命令：

- a. `systemctl stop mediator-scst`
- b. `make scstadm_uninstall`
- c. `make iscsi_uninstall`
- d. `make usr_uninstall`
- e. `make scst_uninstall`
- f. `depmod`

步骤 2：安装 SCST

要手动安装 SCST，您需要用于已安装版本的 ONTAP Mediator 的 SCST tar 捆绑包（请参阅 [SCST 表](#)）。

关于此任务

- 在安装 ONTAP Mediator 之前，请执行此步骤。如果您使用的 SCST 版本高于 ONTAP Mediator 安装程序捆绑的版本，则安装程序将跳过此步骤。
- 在某些情况下，由于升级后的操作系统需要修补程序，因此在升级后的操作系统上默认安装 SCST 可能会失败。

从 ONTAP Mediator 1.12.1 开始，如果安装失败，您可以使用位于 ``usr/local/bin/mediator_repair_scst`` 的 SCST 修复工具来修复安装。

如果升级后的操作系统需要修补程序，则该工具将包含在 ONTAP Mediator 软件包中附带的 SCST 版本中。该 ``mediator_repair_scst`` 工具是一个独立的工具，与主安装程序分开，并打包在 ONTAP Mediator ``tgz`` 文件中。它不会影响 ONTAP Mediator 的安装或升级路径。

步骤

1. 在 `scst` 目录中运行以下 `install` 命令：

- a. `make 2release`
- b. `make scst_install`
- c. `make usr_install`
- d. `make iscsi_install`
- e. `make scstadm_install`
- f. `depmod`



如果您要执行首次安装并希望预安装 ONTAP Mediator，请在继续执行下一步之前运行以下命令：

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

- g. `cp scst/src/certs/scst_module_key.der /opt/netapp/lib/ontap_mediator/ontap_mediator/SCST_mod_keys/`
- h. `patch /etc/init.d/scst < /opt/netapp/lib/ontap_mediator/systemd/scst.patch`



如果在首次安装期间在 ONTAP Mediator 之前预安装 SCST，请跳过此步骤。安装程序将应用相关的 SCST 修补程序。

2. (可选) 如果启用了安全启动，请在重新启动之前执行以下步骤：

- a. 确定 `scst_vdisk`、`scst` 和 `iscsi_scst` 模块的每个文件名：

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

- b. 确定内核版本：

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

- c. 使用内核对每个模块文件进行签名：

```
[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. 使用固件安装 UEFI 密钥。

安装 UEFI 密钥的说明位于：

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

生成的 UEFI 密钥位于：

```
/opt/netapp/lib/ontap_mediator/ontap_mediator/SCST_mod_keys/scst_module_key.de
r
```

3. 重新启动系统：

```
reboot
```

主机更改主机名或 IP

关于此任务

- 在已安装 ONTAP Mediator 的 Linux 主机上执行此任务。
- 仅在安装 ONTAP Mediator 后，主机名或 IP 地址发生变化，因此自签名证书不再有效时才执行此操作。
- 将临时自签名证书替换为受信任的第三方证书后，请不要使用此任务重新生成证书。如果没有自签名证书，则无法使用此过程。

步骤

为当前主机创建临时自签名证书：

1. 重新启动 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
```

卸载 ONTAP Mediator

如有必要，请在断开与 ONTAP 的连接后卸载 ONTAP Mediator。

开始之前

必须先断开 ONTAP Mediator 与 ONTAP 的连接，然后才能将其删除。

关于此任务

在已安装 ONTAP Mediator 的 Linux 主机上执行此任务。

如果无法访问此命令，则可能需要使用完整路径运行该命令，如以下示例中所示：

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

步骤

1. 卸载 ONTAP Mediator:

```
uninstall_ontap_mediator
```

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

ONTAP Mediator: Self Extracting Uninstaller

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

法律声明

法律声明提供对版权声明、商标、专利等信息的访问。

版权

["https://www.netapp.com/company/legal/copyright/"](https://www.netapp.com/company/legal/copyright/)

商标

NETAPP、NETAPP 标识和 NetApp 商标页面上所列的商标是 NetApp, Inc. 的商标。其他公司和产品名称可能是其各自所有者的商标。

["https://www.netapp.com/company/legal/trademarks/"](https://www.netapp.com/company/legal/trademarks/)

专利

当前 NetApp 拥有的专利列表可以在以下网址找到：

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

隐私政策

["https://www.netapp.com/company/legal/privacy-policy/"](https://www.netapp.com/company/legal/privacy-policy/)

开源

通知文件提供有关 NetApp 软件中使用的第三方版权和许可的信息。

用于 MetroCluster IP 和 SnapMirror 主动同步的 ONTAP 调解器

- ["ONTAP Mediator 1.12.1 通知"](#)
- ["ONTAP Mediator 1.11 通知"](#)
- ["ONTAP Mediator 1.10 的通知"](#)
- ["ONTAP Mediator 1.9.1 通知"](#)
- ["ONTAP Mediator 1.8 注意事项"](#)
- ["ONTAP Mediator 1.7 注意事项"](#)
- ["ONTAP Mediator 1.5 注意事项"](#)

版权信息

版权所有 © 2026 NetApp, Inc.。保留所有权利。中国印刷。未经版权所有者事先书面许可，本文档中受版权保护的任何部分不得以任何形式或通过任何手段（图片、电子或机械方式，包括影印、录音、录像或存储在电子检索系统中）进行复制。

从受版权保护的 NetApp 资料派生的软件受以下许可和免责声明的约束：

本软件由 NetApp 按“原样”提供，不含任何明示或暗示担保，包括但不限于适销性以及针对特定用途的适用性的隐含担保，特此声明不承担任何责任。在任何情况下，对于因使用本软件而以任何方式造成的任何直接性、间接性、偶然性、特殊性、惩罚性或后果性损失（包括但不限于购买替代商品或服务；使用、数据或利润方面的损失；或者业务中断），无论原因如何以及基于何种责任理论，无论出于合同、严格责任或侵权行为（包括疏忽或其他行为），NetApp 均不承担责任，即使已被告知存在上述损失的可能性。

NetApp 保留在不另行通知的情况下随时对本文档所述的任何产品进行更改的权利。除非 NetApp 以书面形式明确同意，否则 NetApp 不承担因使用本文档所述产品而产生的任何责任或义务。使用或购买本产品不表示获得 NetApp 的任何专利权、商标权或任何其他知识产权许可。

本手册中描述的产品可能受一项或多项美国专利、外国专利或正在申请的专利的保护。

有限权利说明：政府使用、复制或公开本文档受 DFARS 252.227-7013（2014 年 2 月）和 FAR 52.227-19（2007 年 12 月）中“技术数据权利 — 非商用”条款第 (b)(3) 条规定的限制条件的约束。

本文档中所含数据与商业产品和/或商业服务（定义见 FAR 2.101）相关，属于 NetApp, Inc. 的专有信息。根据本协议提供的所有 NetApp 技术数据和计算机软件具有商业性质，并完全由私人出资开发。美国政府对这些数据的使用权具有非排他性、全球性、受限且不可撤销的许可，该许可既不可转让，也不可再许可，但仅限在与交付数据所依据的美国政府合同有关且受合同支持的情况下使用。除本文档规定的情形外，未经 NetApp, Inc. 事先书面批准，不得使用、披露、复制、修改、操作或显示这些数据。美国政府对国防部的授权仅限于 DFARS 的第 252.227-7015(b)（2014 年 2 月）条款中明确的权利。

商标信息

NetApp、NetApp 标识和 <http://www.netapp.com/TM> 上所列的商标是 NetApp, Inc. 的商标。其他公司和产品名称可能是其各自所有者的商标。