ONTAP SAN Host Utilities

ONTAP SAN Host Utilities

NetApp
January 30, 2026

This PDF was generated from https://docs.netapp.com/us-en/ontap-sanhost/index.html on January 30, 2026. Always check docs.netapp.com for the latest.

Table of Contents

ONTAP SAN Host Utilities . . . 1
Learn about SAN host configurations . . . 2
Install SAN Host Utilities . . . 3
- Learn about SAN Host Utilities . . . 3
- AIX Host Utilities . . . 3
  - AIX Host Utilities Release Notes . . . 3
  - Install AIX Host Utilities . . . 4
  - Learn about the AIX SAN Toolkit for ONTAP storage . . . 12
  - Use AIX Host Utilities commands to verify ONTAP storage configuration . . . 12
- HP-UX Host Utilities . . . 22
  - HP-UX Host Utilities Release Notes . . . 22
  - Install HP-UX Host Utilities 6.0 for ONTAP storage . . . 23
  - Learn about the SAN Toolkit for ONTAP storage . . . 24
  - Use HP-UX Host Utilities commands to verify ONTAP storage configuration . . . 24
- Linux Host Utilities . . . 30
  - Linux Host Utilities Release Notes . . . 30
  - Install Linux Host Utilities . . . 32
  - Learn about the "sanlun" utility for ONTAP storage . . . 36
  - Use Linux Host Utilities commands to verify ONTAP storage configuration . . . 37
- Solaris Host Utilities . . . 44
  - Solaris Host Utilities Release Notes . . . 44
  - Install Solaris Host Utilities . . . 45
  - Learn about the SAN Toolkit for ONTAP storage . . . 50
  - Use Solaris Host Utilities commands to verify ONTAP storage configuration . . . 51
- Windows Host Utilities . . . 56
  - Windows Host Utilities Release Notes . . . 56
  - Install Windows Host Utilities . . . 58
  - Review the Windows Host Utilities configuration for ONTAP storage . . . 66
  - Upgrade Windows Host Utilities for ONTAP storage . . . 78
  - Repair and remove the Windows Host Utilities for ONTAP storage . . . 79
  - Troubleshoot issues for Windows Host Utilities with ONTAP storage configuration . . . 80

Configure hosts with FCP and iSCSI . . . 82
- Overview . . . 82
- AIX and PowerVM/VIOS . . . 82
  - Configure AIX 7.3/VIOS 4.x for FCP and iSCSI with ONTAP storage . . . 82
  - Configure AIX 7.2/VIOS 3.1 for FCP and iSCSI with ONTAP storage . . . 89
  - Configure AIX 7.1 for FCP and iSCSI with ONTAP storage . . . 96
- CentOS . . . 100
  - Configure CentOS 8.x for FCP and iSCSI for ONTAP storage . . . 100
- Citrix . . . 108
  - Configure Citrix Xenserver 8.4 for FCP and iSCSI with ONTAP storage . . . 108
  - Configure Citrix Hypervisor 8.2 for FCP and iSCSI with ONTAP storage . . . 113
- ESXi . . . 118

Use VMware vSphere 8.x with ONTAP . . . 118
Use VMware vSphere 7.x with ONTAP . . . 127
Use VMware vSphere 6.5 and 6.7 with ONTAP . . . 135
HP-UX . . . 142
Configure HP-UX 11i v3 for FCP and iSCSI with ONTAP storage . . . 142
HPE VME . . . 146
Configure HPE VME 8.0.x for FCP and iSCSI with ONTAP storage . . . 147
Oracle Linux . . . 154
Configure Oracle Linux 9.x for FCP and iSCSI with ONTAP storage . . . 154
Configure Oracle Linux 8.x for FCP and iSCSI with ONTAP storage . . . 162
Proxmox . . . 170
Configure Proxmox VE 9.x for FCP and iSCSI with ONTAP storage . . . 170
Configure Proxmox VE 8.x for FCP and iSCSI with ONTAP storage . . . 178
RHEL . . . 186
Configure RHEL 10.x for FCP and iSCSI with ONTAP storage . . . 186
Configure RHEL 9.x for FCP and iSCSI with ONTAP storage . . . 194
Configure RHEL 8.x for FCP and iSCSI with ONTAP storage . . . 204
Rocky Linux . . . 216
Configure Rocky Linux 10.x for FCP and iSCSI with ONTAP storage . . . 217
Configure Rocky Linux 9.x for FCP and iSCSI with ONTAP storage . . . 225
Configure Rocky Linux 8.x for FCP and iSCSI with ONTAP storage . . . 235
Solaris . . . 243
Configure Solaris 11.4 for FCP and iSCSI with ONTAP storage . . . 243
Configure Solaris 11.3 for FCP and iSCSI with ONTAP storage . . . 254
SUSE Linux Enterprise Server . . . 266
Configure SUSE Linux Enterprise Server 16 for FCP and iSCSI with ONTAP storage . . . 266
Configure SUSE Linux Enterprise Server 15 SPx for FCP and iSCSI with ONTAP storage . . . 274
Ubuntu . . . 283
Configure Ubuntu 24.04 for FCP and iSCSI with ONTAP storage . . . 283
Configure Ubuntu 22.04 for FCP and iSCSI with ONTAP storage . . . 291
Configure Ubuntu 20.04 for FCP and iSCSI with ONTAP storage . . . 299
Veritas . . . 307
Configure Veritas Infoscale 9 for FC, FCoE, and iSCSI with ONTAP storage . . . 307
Configure Veritas Infoscale 8 for FC, FCoE, and iSCSI with ONTAP storage . . . 313
Configure Veritas Infoscale 7 for FC, FCoE, and iSCSI with ONTAP storage . . . 319
Configure Veritas Infoscale 6 for FC, FCoE, and iSCSI with ONTAP storage . . . 325
Windows . . . 331
Configure Windows Server 2025 for FCP and iSCSI with ONTAP storage . . . 331
Configure Windows Server 2022 for FCP and iSCSI with ONTAP storage . . . 335
Configure Windows Server 2019 for FCP and iSCSI with ONTAP storage . . . 339
Configure Windows Server 2016 for FCP and iSCSI with ONTAP storage . . . 343
Configure Windows Server 2012 R2 for FCP and iSCSI with ONTAP storage . . . 347
Configure hosts with NVMe-oF . . . 352
Learn about configuring ONTAP SAN hosts with NVMe-oF . . . 352
Configure AIX with NVMe-oF for ONTAP storage . . . 352

Step 1: Confirm the multipath configuration for your host . . . 353
Step 2: Configure NVMe/FC . . . 354
Step 3: Validate NVMe/FC . . . 357
Step 4: Review the known issues . . . 358
Step 5: Troubleshoot . . . 358
ESXi . . . 359
NVMe-oF Host Configuration for ESXi 8.x with ONTAP . . . 359
NVMe-oF Host Configuration for ESXi 7.x with ONTAP . . . 367
Oracle Linux . . . 373
Learn about ONTAP support and features for Oracle Linux hosts . . . 373
Configure Oracle Linux 9.x with NVMe-oF for ONTAP storage . . . 375
Configure Oracle Linux 8.x with NVMe-oF for ONTAP storage . . . 398
Configure Oracle Linux 7.x with NVMe-oF for ONTAP storage . . . 414
Proxmox . . . 421
Learn about ONTAP support and features for Proxmox hosts . . . 422
Configure Proxmox VE 9.x for NVMe-oF with ONTAP storage . . . 422
Configure Proxmox VE 8.x for NVMe-oF with ONTAP storage . . . 438
RHEL . . . 454
Learn about ONTAP support and features for RHEL hosts . . . 455
Configure RHEL 10.x for NVMe-oF with ONTAP storage . . . 456
Configure RHEL 9.x for NVMe-oF with ONTAP storage . . . 480
Configure RHEL 8.x for NVMe-oF with ONTAP storage . . . 506
Rocky Linux . . . 519
Learn about ONTAP support and features for Rocky Linux . . . 519
Configure Rocky Linux 10.x for NVMe-oF with ONTAP storage . . . 520
Configure Rocky Linux 9.x for NVMe-oF with ONTAP storage . . . 543
Configure Rocky Linux 8.x for NVMe-oF with ONTAP storage . . . 571
SUSE Linux Enterprise Server . . . 590
Learn about ONTAP support and features for SUSE Linux Enterprise Server . . . 590
Configure SUSE Linux Enterprise Server 16 for NVMe-oF with ONTAP storage . . . 591
Configure SUSE Linux Enterprise Server 15 SPx for NVMe-oF with ONTAP storage . . . 626
Ubuntu . . . 660
NVMe-oF host configuration for Ubuntu 24.04 with ONTAP . . . 660
NVMe-oF host configuration for Ubuntu 24.04 with ONTAP . . . 675
Windows . . . 685
Configure Windows Server 2025 with NVMe/FC for ONTAP . . . 685
Configure Windows Server 2022 with NVMe/FC for ONTAP . . . 690
NVMe/FC Host Configuration for Windows Server 2019 with ONTAP . . . 696
Configure Windows Server 2016 with NVMe/FC for ONTAP . . . 702
Configure Windows Server 2012 R2 with NVMe/FC for ONTAP . . . 708
Troubleshoot NVMe-oF with ONTAP storage for Linux Host Utilities . . . 714
Enable verbose logging . . . 714
Common "nvme-cli" errors and workarounds . . . 716
When to contact technical support . . . 718
Legal notices . . . 719

Copyright 719
Trademarks 719
Patents 719
Privacy policy 719

ONTAP SAN Host Utilities

Learn about SAN host configurations

You should configure SAN hosts for multipathing and with specific parameters and settings critical for correct operation with ONTAP LUNs. The host configuration varies depending on the host operating system and the protocol. Correct configuration is important for best performance and successful failover.

You should configure your SAN hosts to use the NetApp SAN Host Utilities. SAN Host Utilities help you manage and monitor your LUNs and host bus adapters (HBAs). For example, you can use the SAN Host Utilities to retrieve a list of all LUNs mapped to a host or to see a list of all the attributes of a specific LUN mapped to a host. NetApp customer support can also use the SAN Host Utilities to gather information about your configuration if you have an issue that requires assistance.

NetApp recommends using the SAN Host Utilities software if it's available for your SAN host.

If your SAN configuration supports SAN booting, you can also configure your host to use SAN booting to simplify deployment and improve scalability. Using SAN boot, you can deploy multiple new servers by connecting them to SAN and configuring the appropriate boot settings. This saves time by eliminating the need to install the operating system on each server individually.

Your SAN hosts and ONTAP storage can reside either on-premises, in the cloud, or both.

In a cloud environment, hosts are typically referred to as clients. All host-specific information in this content also applies to cloud clients.

For example, you can configure on-premises ONTAP storage to connect to on-premises SAN hosts or SAN cloud clients. You can also configure storage services running on a cloud environment such as Cloud Volumes ONTAP or Amazon FSx for NetApp ONTAP to connect to on-premises SAN hosts or SAN cloud clients.

Related information

- ONTAP SAN configuration reference
- ONTAP SAN storage management
- ONTAP 9 Release Notes
- SANtricity related documentation in the E-Series systems documentation

Install SAN Host Utilities

Learn about SAN Host Utilities

Learn about the latest information on SAN Host Utility releases and access the installation procedure for your host configuration.

There are several SAN Host Utilities software packages available on the NetApp Support site. NetApp recommends using the SAN Host Utilities software if it's available for your SAN host.

The AIX, HP-UX, Linux, and Solaris Host Utilities software provide a command line toolkit. The toolkit helps you manage ONTAP LUNs and host bus adapters (HBAs). NetApp customer support can also use the SAN Host Utilities to gather information about your configuration if you have an issue that requires assistance. The toolkit is automatically installed when you install the Host Utilities software package.

The Windows Host Utilities enable you to connect a Windows host computer to NetApp storage systems. The software includes an installation program that sets the required Windows registry and HBA parameters. This enables the Windows host to correctly handle the storage system behaviors for ONTAP and E-Series platforms.

For reliable operation after installation, use the Interoperability Matrix Tool to verify that your host supports the complete NVMe over Fabrics (including NVMe over TCP and NVMe over Fibre Channel), iSCSI, FC, or FCoE configuration.

AIX Host Utilities

AIX Host Utilities Release Notes

The release notes describe new features and enhancements, issues fixed in the current release, known issues and limitations, and important cautions related to configuring and managing your specific AIX host with your ONTAP storage system.

For specific information about the operating system versions and updates that the Host Utilities support, see the Interoperability Matrix Tool.

The AIX Host Utilities releases contain the following new features and enhancements.

What's new in AIX Host Utilities 8.0

An IBM Interim Fix (ifix) is available for AIX Host Utilities 8.0 that ensures the `sanlun fcp show adapter -v` command displays the correct HBA speed information, such as supported and negotiated speeds for the adapters. You can install the ifix on the following AIX and VIOS versions:

- AIX: 7.3 TL3 SP0, 7.3 TL2 SP2, 7.3 TL1 SP4, 7.2 TL5 SP9, 7.2 TL5 SP8
- VIOS: 4.1.1.0, 4.1.0.21, 3.1.4.50, 3.1.4.41

What's new in AIX Host Utilities 7.0

AIX Host Utilities 7.0 supports SCSI UNMAP on the AIX host OS. With AIX Host Utilities 7.0, SCSI thin provisioning functions seamlessly with AIX and NetApp LUNs for FC devices.

What's new in AIX Host Utilities 6.1

AIX Host Utilities 6.1 includes support for the memory fault issue that occurred in earlier versions of the AIX host OS. With AIX Host Utilities 6.1, only the sanlun binary has changed. The Microsoft Multipath I/O (MPIO) and related ODM remain unchanged.

Known problems and limitations

You should be aware of the following known problems and limitations that might impact performance on your specific host:

Bug ID	Affects version	Title	Description
N/A	7.0	SCSI UNMAP support for ONTAP iSCSI LUNs	AIX Host Utilities 7.0 only supports the SCSI UNMAP feature for FC devices. SCSI UNMAP support for iSCSI devices are not included in this release.
1069147		AIX HU Sanlun reports incorrect HBA speed	Instances of sanlun displaying incorrect HBA speeds are reported while running the `sanlun fcp show adapter -v` command. The `sanlun fcp show adapter -v` command displays the HBA cards information, such as supported and negotiated speeds for the adapters. This seems to be a reporting issue only. To identify the actual speed, use the `fcstat fcsx` command.

NetApp Bugs Online provides complete information for most known issues, including suggested workarounds where possible. Some keyword combinations and bug types that you might want to use include the following:

- FCP General: Displays FC and HBA bugs that are not associated with a specific host.
- FCP - AIX

What's next?

Learn about installing AIX Host Utilities

Install AIX Host Utilities

Install AIX Host Utilities 8.0 for ONTAP storage

The AIX Host Utilities help you manage ONTAP storage attached to an AIX host. NetApp strongly recommends installing the AIX Host Utilities to improve ONTAP storage

management and assist NetApp support in gathering information about your configuration.

AIX Host Utilities 8.0 supports the following transport protocols and AIX environments:

- FC, FCoE, and iSCSI
- AIX Multipath I/O (MPIO)
- PowerVM

  For information about PowerVM, see the IBM PowerVM Live Partition Mobility Red Book.

About this task

- You need to install the AIX Host Utilities SAN Toolkit with AIX MPIO to manage ONTAP storage attached to an AIX host.
- When you install AIX Host Utilities, they don't change any settings on your AIX host.

Before you begin

- Use the Interoperability Matrix Tool to verify that your AIX OS, protocol, and ONTAP version support SAN booting.
- Enable dynamic tracking for all FC and FCoE initiators by running the `chdev -l <fscsi_device> -a dyntrk=ye` command on the AIX host.

Steps

1. Log in to your host:

   AIX

   On an AIX host, log in as **root**.

   PowerVM

   On a PowerVM host:

   a. Log in as **padmin**.

   b. Become a root user:

   ```
   oem_setup_env
   ```

2. Go to the NetApp Support Site and download the compressed file containing the Host Utilities to a directory on your host.
3. Go to the directory containing the download.
4. Decompress the file and extract the SAN Toolkit software package:

   ```
   tar -xvf ntap_aix_host_utilities_8.0.tar.gz
   ```

   The `ntap_aix_host_utilities_8.0.` directory is created when you decompress the file. This

directory contains one of the following subdirectories: MPIO, NON_MPIO, or SAN_Tool_Kit.

5. Install AIX MPIO:

```
installp -aXYd /var/tmp/ntap_aix_host_utilities_8.0/MPIO
NetApp.MPIO_Host_Utilities_Kit
```

6. Install the SAN Toolkit:

```
installp -aXYd /var/tmp/ntap_aix_host_utilities_8.0/ NetApp.SAN_toolkit
```

7. Reboot the host.
8. Verify the installation:

```
lslpp -l |grep -i netapp
```

Show example output

```
NetApp.MPIO_Host_Utilities_Kit.config
                             8.0.0.0  COMMITTED  NetApp MPIO PCM
Host Utilities
  NetApp.MPIO_Host_Utilities_Kit.fcp
                             8.0.0.0  COMMITTED  NetApp MPIO PCM
Host Utilities
  NetApp.MPIO_Host_Utilities_Kit.iscsi
                             8.0.0.0  COMMITTED  NetApp MPIO PCM
Host Utilities
  NetApp.MPIO_Host_Utilities_Kit.pcmodm
                             8.0.0.0 COMMITTED  NetApp MPIO PCM Host
Utilities
NetApp.SAN_toolkit.sanlun  8.0.0.0 COMMITTED NetApp SAN Toolkit
sanlun
```

9. Confirm that the software version is 8.0.1f0fc74c:

```
sanlun version
```

10. Verify that the SCSI UNMAP `lbp_enabled` parameter has been added to the ODM:

```
odmget -q "uniquetype=disk/fcp/NetAppMPIO" PdAt |grep  "lbp_enabled"
```

Example ouptut

```
attribute = "lbp_enabled"
```

```
odmget -q "uniquetype=disk/fcp/NetAppMPIO and attribute=lbp_enabled"
PdAt`
```

Show example output

```
PdAt:
        uniquetype = "disk/fcp/NetAppMPIO"
        attribute = "lbp_enabled"
        deflt = "true"
        values = "true,false"
        width = ""
        type = "R"
        generic = ""
        rep = "s"
        nls_index = 18
```

What's next?

Learn about the AIX SAN Toolkit.

Install AIX Host Utilities 7.0 for ONTAP storage

The AIX Host Utilities help you manage ONTAP storage attached to an AIX host. NetApp strongly recommends installing the AIX Host Utilities to improve ONTAP storage management and assist NetApp support in gathering information about your configuration.

AIX Host Utilities 7.0 support the following transport protocols and AIX environments:

- FC, FCoE, and iSCSI
- AIX Microsoft Multipath I/O (MPIO)
- PowerVM

  For information about PowerVM, see the IBM PowerVM Live Partition Mobility Red Book.

About this task

- You need to install the AIX Host Utilities SAN Toolkit with AIX MPIO to manage ONTAP storage attached to an AIX host.
- When you install AIX Host Utilities, they don't change any settings on your AIX host.

Before you begin

- Use the Interoperability Matrix Tool to verify that your AIX OS, protocol, and ONTAP version support SAN booting.
- Enable dynamic tracking for all FC and FCoE initiators by running the `chdev -l <fscsi_device> -a dyntrk=ye` command on the AIX host.

Steps

1. Log in to your host.

   AIX

   On an AIX host, log in as **root**.

   PowerVM

   On a PowerVM host:

   a. Log in as **padmin**.

   b. Become a root user:

   ```
   oem_setup_env
   ```

2. Go to the NetApp Support Site and download the compressed file containing the Host Utilities to a directory on your host.
3. Go to the directory containing the download.
4. Decompress the file and extract the SAN Toolkit software package:

   ```
   tar -xvf ntap_aix_host_utilities_7.0.tar.gz
   ```

   The following directory is created when you decompress the file: `ntap_aix_host_utilities_7.0`. This directory contains one of the following subdirectories: MPIO, NON_MPIO, or SAN_Tool_Kit.

5. Install the AIX MPIO:

   ```
   installp -aXYd /var/tmp/ntap_aix_host_utilities_7.0/MPIO
   NetApp.MPIO_Host_Utilities_Kit
   ```

6. Install the SAN Toolkit:

   ```
   installp -aXYd
   /var/tmp/ntap_aix_host_utilities_7.0/SAN_Tool_Kit/NetApp.SAN_toolkit
   ```

7. Reboot the host.

8. Verify the installation:

```
lslpp -l |grep -i netapp
```

```
NetApp.MPIO_Host_Utilities_Kit.config
                             7.0.0.0  COMMITTED  NetApp MPIO PCM Host
Utilities
  NetApp.MPIO_Host_Utilities_Kit.fcp
                             7.0.0.0  COMMITTED  NetApp MPIO PCM Host
Utilities
  NetApp.MPIO_Host_Utilities_Kit.iscsi
                             7.0.0.0  COMMITTED  NetApp MPIO PCM Host
Utilities
  NetApp.MPIO_Host_Utilities_Kit.pcmodm
                             7.0.0.0  COMMITTED  NetApp MPIO PCM Host
Utilities
NetApp.SAN_toolkit.sanlun  7.0.0.0  COMMITTED  NetApp SAN Toolkit sanlun
```

9. Confirm the software version:

```
sanlun version
```

```
7.0.725.3521
```

10. Verify that the SCSI UNMAP `lbp_enabled` parameter is added to the ODM:

```
odmget -q "uniquetype=disk/fcp/NetAppMPIO" PdAt |grep  "lbp_enabled"
```

```
attribute = "lbp_enabled"
```

```
odmget -q "uniquetype=disk/fcp/NetAppMPIO and attribute=lbp_enabled"
PdAt`
```

```
PdAt:
        uniquetype = "disk/fcp/NetAppMPIO"
        attribute = "lbp_enabled"
        deflt = "true"
        values = "true,false"
        width = ""
        type = "R"
        generic = ""
        rep = "s"
        nls_index = 18
```

What's next?

Learn about the AIX SAN Toolkit.

Install AIX Host Utilities 6.1 for ONTAP storage

The AIX Host Utilities help you manage ONTAP storage attached to an AIX host. NetApp strongly recommends installing the AIX Host Utilities to improve ONTAP storage management and assist NetApp support in gathering information about your configuration.

AIX Host Utilities 6.1 support the following transport protocols and AIX environments:

- FC, FCoE, and iSCSI
- AIX Microsoft Multipath I/O (MPIO)
- PowerVM

  For more information about PowerVM, see the IBM PowerVM Live Partition Mobility Red Book.

About this task

- You need to install the AIX Host Utilities SAN Toolkit with AIX MPIO to manage ONTAP storage attached to an AIX host.
- When you install AIX Host Utilities, they don't change any settings on your AIX host.

Before you begin

- Use the Interoperability Matrix Tool to verify that your AIX OS, protocol, and ONTAP version support SAN booting.
- Enable dynamic tracking for all FC and FCoE initiators by running the `chdev -l <fscsi_device> -a dyntrk=ye` command on the AIX host.

Steps

1. Log in to your host.

AIX

On an AIX host, log in as **root**.

PowerVM

On a PowerVM host:

a. Log in as **padmin**.

b. Become a root user:

```
oem_setup_env
```

2. Go to the NetApp Support Site and download the compressed file containing the Host Utilities to a directory on your host.
3. Go to the directory containing the download.
4. Decompress the file and extract the SAN Toolkit software package.

```
tar -xvf ntap_aix_host_utilities_6.1.tar.gz
```

The following directory is created when you decompress the file: `ntap_aix_host_utilities_6.1`. This directory will have one of the following subdirectories: MPIO, NON_MPIO, or SAN_Tool_Kit.

5. Install the AIX MPIO:

```
installp -aXYd /var/tmp/ntap_aix_host_utilities_6.1/MPIO
NetApp.MPIO_Host_Utilities_Kit
```

6. Install the SAN Toolkit:

```
installp -aXYd /var/tmp/ntap_aix_host_utilities_6.1/SAN_Tool_Kit
NetApp.SAN_toolkit
```

7. Reboot the host.
8. Verify the installation:

```
sanlun version
```

What's next?

Learn about the AIX SAN Toolkit.

Learn about the AIX SAN Toolkit for ONTAP storage

AIX Host Utilities is a NetApp host software that provides a command line toolkit on your IBM AIX host. The SAN toolkit is installed when you install the NetApp Host Utilities package. The toolkit provides the `sanlun` utility, which helps you manage ONTAP LUNs and host bus adapters (HBAs). The `sanlun` command returns information about ONTAP LUNs mapped to your host, multipathing, and information necessary to create initiator groups.

The following example output shows the ONTAP LUN information returned for the `sanlun lun show` command:

```
controller(7mode)/ device host lun

vserver(Cmode) lun-pathname filename adapter protocol size mode
--------------------------------------------------------------
data_vserver   /vol/vol1/lun1 hdisk0 fcs0    FCP      60g C
data_vserver   /vol/vol2/lun2 hdisk0 fcs0    FCP      20g C
data_vserver   /vol/vol3/lun3 hdisk11 fcs0   FCP      20g C
data_vserver   /vol/vol4/lun4 hdisk14 fcs0   FCP      20g C
```

> The SAN Toolkit is common across all AIX host configurations and transport protocols. As a result, all of the components don't apply to every configuration. Unused components don't affect your system performance. The SAN Toolkit is supported on AIX and PowerVM/VIOS OS versions.

What's next?

Learn about using the AIX Host Utilities tool.

Use AIX Host Utilities commands to verify ONTAP storage configuration

You can use the AIX Host Utilities sample command reference for an end-to-end validation of the NetApp storage configuration using the Host Utilities tool.

List all host initiators mapped to host

You can retrieve a list of host initiators mapped to a host.

```
sanlun fcp show adapter -v
```

8.0

Show example for AIX Host Utilities 8.0

```
adapter name:      fcs4
WWPN:              100000109bf606a8
WWNN:              200000109bf606a8
driver name:       /usr/lib/drivers/pci/emfcdd
model:             df1000e31410150
model description: FC Adapter
serial number:     Y050HY22L002
hardware version:  Not Available
driver version:    7.2.5.201
firmware version:  00014000000057400007
Number of ports:   1
port type:         Fabric
port state:        Operational
supported speed:   32 GBit/sec
negotiated speed:  32 GBit/sec
OS device name:    fcs4
adapter name:      fcs5
WWPN:              100000109bf606a9
WWNN:              200000109bf606a9
driver name:       /usr/lib/drivers/pci/emfcdd
model:             df1000e31410150
model description: FC Adapter
serial number:     Y050HY22L002
hardware version:  Not Available
driver version:    7.2.5.201
firmware version:  00014000000057400007
Number of ports:   1
port type:         Fabric
port state:        Operational
supported speed:   32 GBit/sec
negotiated speed:  32 GBit/sec
OS device name:    fcs5
bash-3.2#
```

7.0 and 6.1

Show example for AIX Host Utilities 7.0 and 6.1

```
bash-3.2# sanlun fcp show adapter -v
adapter name: fcs0
WWPN: 100000109b22e143
WWNN: 200000109b22e143
driver name: /usr/lib/drivers/pci/emfcdd
model: df1000e31410150
model description: FC Adapter
serial number: YA50HY79S117
hardware version: Not Available
driver version: 7.2.5.0
firmware version: 00012000040025700027
Number of ports: 1
port type: Fabric
port state: Operational
supported speed: 16 GBit/sec
negotiated speed: Unknown
OS device name: fcs0
adapter name: fcs1
WWPN: 100000109b22e144
WWNN: 200000109b22e144
driver name: /usr/lib/drivers/pci/emfcdd
model: df1000e31410150
model description: FC Adapter
serial number: YA50HY79S117
hardware version: Not Available
driver version: 7.2.5.0
firmware version: 00012000040025700027
Number of ports: 1
port type: Fabric
port state: Operational
supported speed: 16 GBit/sec
negotiated speed: Unknown
OS device name: fcs1
bash-3.2#
```

List all LUNs mapped to host

You can retrieve a list of all LUNs mapped to a host.

```
sanlun lun show -p -v all
```

8.0

Show example for AIX Host Utilities 8.0

```
LUN: 88
LUN Size: 15g
Host Device: hdisk9
Mode: C
Multipath Provider: AIX Native
Multipathing Algorithm: round_robin

host    vserver  AIX   AIX MPIO
path    path     MPIO  host      vserver      path
state   type     path  adapter   LIF          priority
-------------------------------------------------------
up     primary   path0   fcs0    fc_aix_1     1
up     primary   path1   fcs1    fc_aix_2     1
up     secondary path2   fcs0    fc_aix_3     1
up     secondary path3   fcs1    fc_aix_4     1
```

7.0 and 6.1

Show example for AIX Host Utilities 7.0 and 6.1

```
ONTAP Path:
vs_aix_clus:/vol/gpfs_205p2_207p1_vol_0_8/aix_205p2_207p1_lun
LUN: 88
LUN Size: 15g
Host Device: hdisk9
Mode: C
Multipath Provider: AIX Native
Multipathing Algorithm: round_robin

host    vserver  AIX   AIX MPIO
path    path     MPIO  host      vserver      path
state   type     path  adapter   LIF          priority
-------------------------------------------------------
up     primary   path0   fcs0    fc_aix_1     1
up     primary   path1   fcs1    fc_aix_2     1
up     secondary path2   fcs0    fc_aix_3     1
up     secondary path3   fcs1    fc_aix_4     1
```

List all LUNs mapped to host from a given SVM

You can retrieve a list of all LUNs mapped to a host from a specified SVM.

```
sanlun lun show -p -v sanboot_unix
```

8.0

Show example for AIX Host Utilities 8.0

```
ONTAP Path: sanboot_unix:/vol/aix_205p2_boot_0/boot_205p2_lun
LUN: 0
LUN Size: 80.0g
Host Device: hdisk85
Mode: C
Multipath Provider: AIX Native
Multipathing Algorithm: round_robin

host    vserver    AIX   AIX MPIO
path    path       MPIO  host    vserver    path
state   type       path  adapter LIF        priority
-------------------------------------------------
up      primary    path0 fcs0    sanboot_1   1
up      primary    path1 fcs1    sanboot_2   1
up      secondary  path2 fcs0    sanboot_3   1
up      secondary  path3 fcs1    sanboot_4   1
```

7.0 and 6.1

Show example for AIX Host Utilities 7.0 and 6.1

```
ONTAP Path: sanboot_unix:/vol/aix_205p2_boot_0/boot_205p2_lun
LUN: 0
LUN Size: 80.0g
Host Device: hdisk85
Mode: C
Multipath Provider: AIX Native
Multipathing Algorithm: round_robin

host    vserver    AIX   AIX MPIO
path    path       MPIO  host    vserver    path
state   type       path  adapter LIF        priority
-------------------------------------------------
up      primary    path0 fcs0    sanboot_1   1
up      primary    path1 fcs1    sanboot_2   1
up      secondary  path2 fcs0    sanboot_3   1
up      secondary  path3 fcs1    sanboot_4   1
```

List all attributes of a given LUN mapped to host

You can retrieve a list of all attributes of a specified LUN mapped to a host.

```
sanlun lun show -p -v
vs_aix_clus:/vol/gpfs_205p2_207p1_vol_0_8/aix_205p2_207p1_lun
```

8.0

Show example for AIX Host Utilities 8.0

```
ONTAP Path:
vs_aix_clus:/vol/gpfs_205p2_207p1_vol_0_8/aix_205p2_207p1_lun
LUN: 88
LUN Size: 15g
Host Device: hdisk9
Mode: C
Multipath Provider: AIX Native
Multipathing Algorithm: round_robin

host     vserver   AIX   AIX MPIO
path     path      MPIO  host     vserver   path
state    type      path  adapter  LIF       priority
-------------------------------------------------------
up       primary   path0 fcs0    fc_aix_1   1
up       primary   path1 fcs1    fc_aix_2   1
up       secondary path2 fcs0    fc_aix_3   1
up       secondary path3 fcs1    fc_aix_4   1
```

7.0 and 6.1

Show example for AIX Host Utilities 7.0 and 6.1

```
ONTAP Path:
vs_aix_clus:/vol/gpfs_205p2_207p1_vol_0_8/aix_205p2_207p1_lun
LUN: 88
LUN Size: 15g
Host Device: hdisk9
Mode: C
Multipath Provider: AIX Native
Multipathing Algorithm: round_robin

host     vserver   AIX   AIX MPIO
path     path      MPIO  host     vserver   path
state    type      path  adapter  LIF       priority
-------------------------------------------------------
up       primary   path0 fcs0    fc_aix_1   1
up       primary   path1 fcs1    fc_aix_2   1
up       secondary path2 fcs0    fc_aix_3   1
up       secondary path3 fcs1    fc_aix_4   1
```

List ONTAP LUN attributes by host device filename

You can retrieve a list of ONTAP LUN attributes by specifying a host device filename.

```
sanlun lun show -d /dev/hdisk1
```

8.0

Show example for AIX Host Utilities 8.0

```
controller(7mode)/
device host lun
vserver(Cmode)     lun-pathname
----------------------------------------------------------------------
----------
vs_aix_clus       /vol/gpfs_205p2_207p1_vol_0_0/aix_205p2_207p1_lun

filename adapter protocol size mode
----------------------------------
hdisk1    fcs0    FCP     15g  C
```

7.0 and 6.1

Show example for AIX Host Utilities 7.0 and 6.1

```
controller(7mode)/
device host lun
vserver(Cmode)     lun-pathname
----------------------------------------------------------------------
----------
vs_aix_clus       /vol/gpfs_205p2_207p1_vol_0_0/aix_205p2_207p1_lun

filename adapter protocol size mode
----------------------------------
hdisk1    fcs0    FCP     15g  C
```

List all SVM target LIF WWPNs attached to host

You can retrieve a list of all SVM target LIF WWPNs attached to a host.

```
sanlun lun show -wwpn
```

8.0

Show example for AIX Host Utilities 8.0

```
controller(7mode)/
target device host lun
vserver(Cmode)          wwpn            lun-pathname
------------------------------------------------------------------------
-------------

vs_aix_clus          203300a098ba7afe
/vol/gpfs_205p2_207p1_vol_0_0/aix_205p2_207p1_lun
vs_aix_clus          203300a098ba7afe
/vol/gpfs_205p2_207p1_vol_0_9/aix_205p2_207p1_lun
vs_aix_clus          203300a098ba7afe
/vol/gpfs_205p2_207p1_vol_en_0_0/aix_205p2_207p1_lun_en
vs_aix_clus          202f00a098ba7afe
/vol/gpfs_205p2_207p1_vol_en_0_1/aix_205p2_207p1_lun_en

filename     adapter    size  mode
----------------------------------
hdisk1       fcs0       15g    C
hdisk10      fcs0       15g    C
hdisk11      fcs0       15g    C
hdisk12      fcs0       15g    C
```

7.0 and 6.1

Show example for AIX Host Utilities 7.0 and 6.1

```
controller(7mode)/
target device host lun
vserver(Cmode)          wwpn            lun-pathname
-------------------------------------------------------------------------
-------------

vs_aix_clus          203300a098ba7afe
/vol/gpfs_205p2_207p1_vol_0_0/aix_205p2_207p1_lun
vs_aix_clus          203300a098ba7afe
/vol/gpfs_205p2_207p1_vol_0_9/aix_205p2_207p1_lun
vs_aix_clus          203300a098ba7afe
/vol/gpfs_205p2_207p1_vol_en_0_0/aix_205p2_207p1_lun_en
vs_aix_clus          202f00a098ba7afe
/vol/gpfs_205p2_207p1_vol_en_0_1/aix_205p2_207p1_lun_en

filename     adapter    size  mode
----------------------------------
hdisk1       fcs0       15g    C
hdisk10      fcs0       15g    C
hdisk11      fcs0       15g    C
hdisk12      fcs0       15g    C
```

HP-UX Host Utilities

HP-UX Host Utilities Release Notes

The release notes describe new features and enhancements, fixed issues, known issues, limitations, and important cautions related to configuring and managing your specific HP-UX host with your ONTAP storage system.

What's new in HP-UX Host Utilities 6.0

There are no new features and enhancements.

HP-UX Host Utilities 6.0 supports the following HP-UX OS versions:

- HP-UX 11iv2
- HP-UX 11iv3

Known issues and limitations

There are no known issues or limitations.

What's next?

Learn about installing HP-UX Host Utilities

Install HP-UX Host Utilities 6.0 for ONTAP storage

The HP-UX Host Utilities help you manage ONTAP storage attached to an HP-UX host. NetApp strongly recommends installing the HP-UX Host Utilities, but it isn't mandatory. The utilities improve management and assist NetApp customer support in gathering information about your configuration.

The HP-UX Host Utilities support the following environments:

- Native Microsoft Multipath I/O (MPIO)
- Veritas Dynamic Multipathing (DMP)

Before you begin

For reliable operation, use the Interoperability Matrix Tool to verify that your iSCSI, FC, or FCoE configuration is supported.

Steps

1. Log in to your HP-UX host host.
2. Download the HP-UX Host Utilities file `netapp_hpux_host_utilities_6.0_ia_pa.depot.gz` from the NetApp Support Site to your HP-UX host.
3. Decompress the `netapp_hpux_host_utilities_6.0_ia_pa.depot.gz` file:

   ```
   gunzip netapp_hpux_host_utilities_6.0_ia_pa.depot.gz
   ```

   The system places the extracted software in the directory where you uncompressed the depot file.

4. Install the software:

   ```
   swinstall -s /depot_path
   ```

   `depot_path` provides the path and name of the depot file.

   The `swinstall` command runs an installation script that verifies the status of your HP-UX setup. If your system meets the requirements, this script installs the `sanlun` utility and diagnostic scripts in the `/opt/NetApp/santools/bin` directory.

5. Verify the installation:

   ```
   sanlun version
   ```

What's next?

Learn about the SAN Toolkit.

Learn about the SAN Toolkit for ONTAP storage

HP-UX Host Utilities is a NetApp host software that provides a command line toolkit on your HP-UX host. The toolkit is installed when you install the NetApp Host Utilities package. This kit provides the `sanlun` utility, which helps you manage ONTAP LUNs and host bus adapters. The `sanlun` command returns information about the LUNs mapped to your host, multipathing, and information necessary to create initiator groups.

In the following example, the `sanlun lun show` command returns ONTAP LUN information.

```
# sanlun lun show all

controller(7mode)/ device host lun
vserver(Cmode)           lun-pathname                    filename
adapter    protocol   size   mode
--------------------------------------------------------------------------
---------------------------------
sanboot_unix         /vol/hpux_boot/boot_hpux_lun      /dev/rdsk/c34t0d0
fclp1      FCP        150g    C
sanboot_unix         /vol/hpux_boot/boot_hpux_lun      /dev/rdsk/c23t0d0
fclp1      FCP        150g    C
sanboot_unix         /vol/hpux_boot/boot_hpux_lun      /dev/rdsk/c12t0d0
fclp0      FCP        150g    C
sanboot_unix         /vol/hpux_boot/boot_hpux_lun      /dev/rdsk/c81t0d0
fclp0      FCP        150g    C
```

This SAN toolkit is common across all Host Utilities configurations and protocols. As a result, all of the components don't apply to every configuration. Unused components don't affect your system performance.

The SAN Toolkit is supported on AIX and PowerVM/VIOS OS versions.

What's next?

Learn about using the HP-UX Host Utilities tool.

Use HP-UX Host Utilities commands to verify ONTAP storage configuration

Use the HP-UX Host Utilities 6.0 sample command reference for an end-to-end validation of the ONTAP storage configuration using the Host Utilities tool.

List all host initiators mapped to host

Retrieve a list of all host initiators mapped to a host.

```
sanlun fcp show adapter -v
```

Show example output

```
adapter name:      fclp2
WWPN:              10000000c985ef92
WWNN:              20000000c985ef92
driver name:       fclp
model:             AJ763-63001
model description: HP 8Gb Dual Channel PCI-e 2.0 FC HBA
serial number:     MY19034N9U
hardware version:  3
driver version:    @(#) FCLP: PCIe Fibre Channel driver (FibrChanl-02),
B.11.31.1805, Feb  5 2018, FCLP_IFC (3,2)
firmware version:  2.02X2 SLI-3 (U3D2.02X2)
Number of ports:   1 of 2
port type:         Unknown
port state:        Link Down
supported speed:   8 GBit/sec
negotiated speed:  Speed not established
OS device name:    /dev/fclp2

adapter name:      fclp3
WWPN:              10000000c985ef93
WWNN:              20000000c985ef93
driver name:       fclp
model:             AJ763-63001
model description: HP 8Gb Dual Channel PCI-e 2.0 FC HBA
serial number:     MY19034N9U
hardware version:  3
driver version:    @(#) FCLP: PCIe Fibre Channel driver (FibrChanl-02),
B.11.31.1805, Feb  5 2018, FCLP_IFC (3,2)
firmware version:  2.02X2 SLI-3 (U3D2.02X2)
Number of ports:   2 of 2
port type:         Unknown
port state:        Link Down
supported speed:   8 GBit/sec
negotiated speed:  Speed not established
OS device name:    /dev/fclp3
```

List all LUNs mapped to host

Retrieve a list of all LUNs mapped to a host.

```
sanlun lun show -p -v all
```

Show example output

```
\
                    ONTAP Path:
vs_hp_cluster:/vol/chathpux_217_vol_en_1_10/hp_en_217_lun
                           LUN: 55
                      LUN Size: 15g
                   Host Device: /dev/rdisk/disk718
                          Mode: C
                            VG: /dev/vg_data
              Multipath Policy: A/A
            Multipath Provider: Native
-------- ---------- ------------------- -------
----------------------------------------------- -------------
host     vserver    /dev/dsk
HP A/A
path     path       filename             host    vserver
path failover
state    type       or hardware path     adapter LIF
priority
-------- ---------- ------------------- -------
----------------------------------------------- -------------
up       primary    /dev/dsk/c37t6d7     fclp0   hpux_7
0
up       primary    /dev/dsk/c22t6d7     fclp1   hpux_8
0
up       secondary  /dev/dsk/c36t6d7     fclp0   hpux_5
1
up       secondary  /dev/dsk/c44t6d7     fclp1   hpux_6
1
```

List all LUNs mapped to host from a given SVM

Retrieve a list of all LUNs mapped to a host from a specific SVM.

```
sanlun lun show -p -v vs_hp_cluster
```

Show example output

```
                    ONTAP Path:
vs_hp_cluster:/vol/chathpux_217_vol_en_1_10/hp_en_217_lun
                           LUN: 55
                      LUN Size: 15g
                   Host Device: /dev/rdisk/disk718
                          Mode: C
                            VG: /dev/vg_data
              Multipath Policy: A/A
            Multipath Provider: Native
-------- ---------- -------------------- -------
----------------------------------------------- -------------
host     vserver    /dev/dsk
HP A/A
path     path       filename             host    vserver
path failover
state    type       or hardware path     adapter LIF
priority
-------- ---------- -------------------- -------
----------------------------------------------- -------------
up       primary    /dev/dsk/c37t6d7     fclp0   hpux_7
0
up       primary    /dev/dsk/c22t6d7     fclp1   hpux_8
0
up       secondary  /dev/dsk/c36t6d7     fclp0   hpux_5
1
up       secondary  /dev/dsk/c44t6d7     fclp1   hpux_6
1
```

List all attributes of a given LUN mapped to host

Retrieve a list of all attributes of a specified LUN mapped to a host.

```
sanlun lun show -p -v
vs_hp_cluster:/vol/chathpux_217_vol_en_1_5/hp_en_217_lun
```

Show example output

```
                    ONTAP Path:
vs_hp_cluster:/vol/chathpux_217_vol_en_1_5/hp_en_217_lun
                           LUN: 49
                      LUN Size: 15g
                   Host Device: /dev/rdisk/disk712
                          Mode: C
                            VG: /dev/vg_data
              Multipath Policy: A/A
            Multipath Provider: Native
-------- ---------- -------------------- -------
---------------------------------------------- -------------
host     vserver    /dev/dsk
HP A/A
path     path       filename             host    vserver
path failover
state    type       or hardware path     adapter LIF
priority
-------- ---------- -------------------- -------
---------------------------------------------- -------------
up       primary    /dev/dsk/c37t6d1     fclp0   hpux_7
0
up       primary    /dev/dsk/c22t6d1     fclp1   hpux_8
0
up       secondary  /dev/dsk/c36t6d1     fclp0   hpux_5
1
up       secondary  /dev/dsk/c44t6d1     fclp1   hpux_6
1
```

List ONTAP LUN attributes by host device filename

Retrieve a list of ONTAP LUN attributes by a specified host device filename.

```
sanlun lun show -dv /dev/rdisk/disk716
```

Show example output

```
                                                                        device
host                  lun
vserver               lun-pathname
filename           adapter    protocol   size    mode
------------------------------------------------------------------------
------------------------------------------
vs_hp_cluster        /vol/chathpux_217_vol_en_1_14/hp_en_217_lun
/dev/rdisk/disk716 0          FCP        15g     C
             LUN Serial number: 80D7l?NiNP5U
         Controller Model Name: AFF-A800
          Vserver FCP nodename: 208400a098ba7afe
          Vserver FCP portname: 207e00a098ba7afe
              Vserver LIF name: hpux_5
            Vserver IP address: 10.141.54.30
                                10.141.54.35
                                10.141.54.37
                                10.141.54.33
                                10.141.54.31
           Vserver volume name: chathpux_217_vol_en_1_14
MSID::0x00000000000000000000000080915935
         Vserver snapshot name:
```

List all SVM target LIF WWPNs attached to host

Retrieve a list of all SVM target LIF WWPNs attached to a host.

```
sanlun lun show -wwpn
```

Show example output

```
controller(7mode)/
vserver(Cmode)        target wwpn          lun-pathname
device filename
------------------------------------------------------------------------
---------------------------------
vs_hp_cluster         208300a098ba7afe
/vol/chathpux_217_vol_en_1_10/hp_en_217_lun   /dev/rdsk/c22t6d7
vs_hp_cluster         208100a098ba7afe
/vol/chathpux_217_vol_en_1_10/hp_en_217_lun   /dev/rdsk/c44t6d7
vs_hp_cluster         208200a098ba7afe
/vol/chathpux_217_vol_en_1_10/hp_en_217_lun   /dev/rdsk/c37t6d7
vs_hp_cluster         207e00a098ba7afe
/vol/chathpux_217_vol_en_1_10/hp_en_217_lun   /dev/rdsk/c36t6d7
vs_hp_cluster         207d00a098ba7afe   /vol/chathpux_217_os/hp_217_os
/dev/rdsk/c18t7d4
vs_hp_cluster         207f00a098ba7afe   /vol/chathpux_217_os/hp_217_os
/dev/rdsk/c42t7d4

host adapter    lun size    mode
---------------------------------
fclp1           15g         C
fclp1           15g         C
fclp0           15g         C
fclp0           15g         C
fclp1           30g         C
fclp0           30g         C
```

Linux Host Utilities

Linux Host Utilities Release Notes

The release notes describe new features, enhancements, fixed issues, known issues, limitations, and important cautions for configuring and managing your specific host with your ONTAP storage system.

For specific information about the operating system versions and updates that the Host Utilities support, see the Interoperability Matrix Tool.

What's New in Linux Host Utilities 8.0

The Linux Host Utilities 8.0 release contains the following new features and enhancements

Linux Host Utilities 8.0 includes support for 64-bit speed QLogic and Emulex FC host bus adapters (HBA).

The following operating systems are supported:

- Red Hat Enterprise Linux (RHEL) 9 and 8 series
- SUSE Linux Enterprise Server
- Oracle Linux 9 and 8 series
- Ubuntu

What's New in Linux Host Utilities 7.1

The Linux Host Utilities 7.1 release contains the following new features and enhancements:

- Linux Host Utilities is now called *Linux Unified Host Utilities* because it supports NetApp E-Series storage systems running SANtricity as well as AFF, FAS, and ASA systems running ONTAP.
- The following operating systems are supported:
  - Citrix XenServer
  - KVM and XEN, RHEV 6.5 and 6.4
  - Oracle VM 3.2 series
  - Oracle Linux 7 and 6 series
  - RHEL 7 and 6 series
  - SUSE Linux Enterprise Server 15 series
  - SUSE Linux Enterprise Server 11 SP4
- On RHEL 6 and 7 hosts, a tuned package for setting server profiles is now supported. You can use the `tuned-adm` command to set different profiles, depending on the environment. For example, you can use the virtual guest profile as a guest virtual machine and the enterprise storage profile for configurations when LUNs from enterprise storage arrays are used. Using these tuned packages can result in improvement in throughput and latency in ONTAP.
- Linux Host Utilities 7.1 adds support for 32GB FC adapters from Broadcom Emulex and Marvell Qlogic.

NetApp continues to work with the Linux Host Utilities to add support for features after the initial release. For latest information about the features that are supported and the new features that have been added, see the Interoperability Matrix Tool.

Fixed issues

The following issues have been fixed in Linux Host Utilities.

Fixed in version	Description
7.1	The intermittent host OS failure issue that occurs when running the `sanlun lun show -p` command in SUSE Linux Enterprise Server 12 SP1, Oracle Linux 7.2, RHEL 7.2, and RHEL 6.8.

Known issues and limitations

You should be aware of the following known issues and limitations that might impact performance on your specific host:

NetApp Bug ID	Affects version	Title	Description

1457017	7.1	`sanlun` installation issues warning messages related to `libdevmapper.so` and `libnl.so` libraries. These warnings do not affect the functionality of `sanlun` kit.	When you execute the Linux Host Utilities CLI command - "sanlun fcp show adapter -v" on a SAN host, the command fails with an error message displaying that the library dependencies required for an host bus adapter HBA discovery cannot be located:
			
			`[root@hostname ~]# sanlun fcp show adapter -v`
			`Unable to locate /usr/lib64/libHBAAPI.so library`
			`Make sure the package installing the library is installed & loaded`
			Refer to NetApp Bugs Online - 1508554.

NetApp Bugs Online provides complete information for most known issues, including suggested workarounds where possible.

What's next

Learn about installing Linux Host Utilities

Install Linux Host Utilities

Install Linux Host Utilities 8.0 for ONTAP storage

The Linux Host Utilities help you manage ONTAP storage attached to a Linux host. NetApp strongly recommends installing the Linux Host Utilities, but it isn't mandatory. The utilities improve management and assist NetApp customer support in gathering information about your configuration.

Linux Host Utilities 8.0 supports the following distribution types:

- Red Hat Enterprise Linux (RHEL)
- SUSE Linux Enterprise Server
- Oracle Linux
- Ubuntu

> The Linux Host Utilities software doesn't support NVMe over Fibre Channel (NVMe/FC) and NVMe over TCP (NVMe/TCP) host protocols.

About this task

When you install the Linux Host Utilities, it doesn't change any settings on your Linux host.

Before you begin

- For reliable operation, use the Interoperability Matrix Tool to verify that your iSCSI, FC, or FCoE configuration is supported.
- Install the host bus adapter (HBA) management packages available on the vendor support site. The management software enables the `sanlun` commands to gather information about the FC HBAs, such as their WWPNs.

  Refer to the vendor documentation to verify that the following packages are correctly installed. These packages are required to support the `sanlun fcp show adapter` command:

  - Marvell QLogic HBA – QConvergeConsole CLI
  - Broadcom Emulex HBA - OneCommand Manager core application CLI

Steps

1. If you have a version of Linux Host Utilities currently installed, remove it:

   Linux hosts

   Remove Linux Host Utilities from a RHEL, SUSE Linux Enterprise Server, or Oracle Linux host:

   ```
   rpm -e netapp_linux_unified_host_utilities-x-x
   ```

   Ubuntu

   Remove Linux Host Utilities from an Ubuntu host:

   ```
   sudo apt remove netapp_linux_unified_host_utilities-x-x
   ```

   For earlier versions of Linux Host Utilities, go to the directory where the host utility software is installed and enter the uninstall command to remove the installed package.

2. The NetApp Linux Host Utilities software package is available on the NetApp Support Site in a 64-bit .rpm file. Download the 64-bit file from the NetApp Support Site to your host.
3. Go to the directory to which you downloaded the software package and install it:

Linux hosts

Install Linux Host Utilities 8.0 on a RHEL, SUSE Linux Enterprise Server, or Oracle Linux host:

```
rpm -ivh netapp_linux_unified_host_utilities-8-0.x86_xx.rpm
```

You should see an output similar to the following example:

```
rpm -ivh netapp_linux_unified_host_utilities-8-0.x86_64.rpm
Verifying...
 ################################# [100%]
Preparing...
 ################################# [100%]
Updating / installing...

 1:netapp_linux_unified_host_utiliti################################
# [100%]
```

Ubuntu

a. Install Linux Host Utilities 8.0 on an Ubuntu host:

```
sudo apt install
/<path_to_file>/netapp_linux_unified_host_utilities-8-
0.x86_xx.deb
```

b. Manually link the Ubuntu OS to the HBA library:

```
cp
/opt/QLogic_Corporation/QConvergeConsoleCLI/lib64/libHBAAPI.so.2.
0.2 /usr/lib64/libHBAAPI.so
```

4. Verify the installation:

```
sanlun version
```

You should see an output similar to the following example:

```
sanlun version 8.0.386.1644
```

What's next?

- Recommended driver settings with Linux kernel

  When you configure an FC environment that uses native inbox drivers that are bundled with the Linux kernel, you can use the default values for the drivers.

- Learn about the "sanlun" utility.

Install Linux Host Utilities 7.1 for ONTAP storage

The Linux Host Utilities help you manage ONTAP storage attached to a Linux host. NetApp strongly recommends installing the Linux Host Utilities, but it isn't mandatory. The utilities improve management and assist NetApp customer support in gathering information about your configuration.

Linux Host Utilities 7.1 supports the following distribution types:

- Red Hat Enterprise Linux
- SUSE Linux Enterprise Server
- Oracle Linux
- Oracle VM
- Citrix XenServer

> The Linux Host Utilities software doesn't support NVMe over Fibre Channel (NVMe/FC) and NVMe over TCP (NVMe/TCP) host protocols.

About this task

When you install the Linux Host Utilities, it doesn't change any settings on your Linux host.

Before you begin

- For reliable operation, use the Interoperability Matrix Tool to verify that your iSCSI, FC, or FCoE configuration is supported.
- Install the host bus adapter (HBA) management packages available on the vendor support site. The management software enables the `sanlun` commands to gather information about the FC HBAs, such as their WWPNs.

  Refer to the vendor documentation to verify that the following packages are correctly installed. These packages are required to support the `sanlun fcp show adapter` command:

  - Marvell QLogic HBA – QConvergeConsole CLI
  - Broadcom Emulex HBA - OneCommand Manager core application CLI
  - Marvell Brocade HBA – Brocade Command Utility CLI
- Install the RPM "libhbaapi" and "libhbalinux" packages available for each Linux distribution on the Linux host OS.

Steps

1. If you have a version of Linux Host Utilities currently installed, remove it:

```
rpm -e netapp_linux_unified_host_utilities-7-1
```

For earlier versions of Linux Host Utilities, go to the directory where the host utility software is installed and enter the uninstall command to remove the installed package.

2. Download the 32-bit or 64-bit Linux Host Utilities software package from the NetApp Support Site to your host.
3. Go to the directory to which you downloaded the software package and install it:

```
rpm -ivh netapp_linux_unified_host_utilities-7-1.x86_xx.rpm
```

You should see an output similar to the following example:

```
Verifying...                          ##################################
[100%]
Preparing...                          ##################################
[100%]
Updating / installing...
   1:netapp_linux_unified_host_utiliti##################################
[100%]
```

4. Verify the installation:

```
sanlun version
```

You should see an output similar to the following example:

```
sanlun version 7.1.386.1644
```

What's next?

- Recommended driver settings with Linux kernel

  When you configure an FC environment that uses native inbox drivers that are bundled with the Linux kernel, you can use the default values for the drivers.

- Learn about the "sanlun" utility.

Learn about the "sanlun" utility for ONTAP storage

Linux Host Utilities is a NetApp host software that provides `sanlun` commands on your Linux host. The `sanlun` utility is installed automatically when you install the NetApp Host Utilities package. This utility provides the `sanlun` commands that you can use to manage

ONTAP LUNs and host bus adapters (HBAs). The `sanlun` commands return information about the ONTAP LUNs mapped to your host, multipathing, and information necessary to create initiator groups.

In the following example, the `sanlun lun show all` command returns ONTAP LUN information:

```
controller(7mode/E-Series)/            device     host                lun
vserver(cDOT/FlashRay)   lun-pathname  filename   adapter  protocol   size
Product
------------------------------------------------------------------------------
----------
data_vserver          /vol/vol1/lun1   /dev/sdb   host16   FCP
120.0g  cDOT
data_vserver          /vol/vol1/lun1   /dev/sdc   host15   FCP
120.0g  cDOT
data_vserver          /vol/vol2/lun2   /dev/sdd   host16   FCP
120.0g  cDOT
data_vserver          /vol/vol2/lun2   /dev/sde   host15   FCP
120.0g  cDOT
```

- For Linux Host Utilities 7.1, the "sanlun" utility is common across all configurations and protocols of the Host Utilities. As a result, some of its contents apply to one configuration, but not another. Having unused components doesn't affect your system performance.
- The "sanlun" utility isn't supported for the following hypervisor types:

8.0

For Linux Host Utilities 8.0, the "sanlun" utility isn't supported for Citrix XenServer, Oracle VM, and Red Hat Enterprise Virtualization.

7.1

For Linux Host Utilities 7.1, the "sanlun" utility isn't supported for Citrix Xenserver, Red hat Enterprise Virtualization, and Proxmox.

What's next?

Learn about using the Linux Host Utilities tool.

Use Linux Host Utilities commands to verify ONTAP storage configuration

You can use the Linux Host Utilities sample command reference for an end-to-end validation of the NetApp storage configuration using the Host Utilities tool.

List all host initiators mapped to host

You can retrieve a list of all host initiators mapped to a host.

```
sanlun fcp show adapter -v
```

Show example

```
adapter name:      host15
WWPN:              10000090fa022736
WWNN:              20000090fa022736
driver name:       lpfc
model:             LPe16002B-M6
model description: Emulex LPe16002B-M6 PCIe 2-port 16Gb Fibre Channel
Adapter
serial number:     FC24637890
hardware version:  0000000b 00000010 00000000
driver version:    12.8.0.5; HBAAPI(I) v2.3.d, 07-12-10
firmware version:  12.8.340.8
Number of ports:   1
port type:         Fabric
port state:        Operational
supported speed:   4 GBit/sec, 8 GBit/sec, 16 GBit/sec
negotiated speed:  16 GBit/sec
OS device name:    /sys/class/scsi_host/host15

adapter name:      host16
WWPN:              10000090fa022737
WWNN:              20000090fa022737
driver name:       lpfc
model:             LPe16002B-M6
model description: Emulex LPe16002B-M6 PCIe 2-port 16Gb Fibre Channel
Adapter
serial number:     FC24637890
hardware version:  0000000b 00000010 00000000
driver version:    12.8.0.5; HBAAPI(I) v2.3.d, 07-12-10
firmware version:  12.8.340.8
Number of ports:   1
port type:         Fabric
port state:        Operational
supported speed:   4 GBit/sec, 8 GBit/sec, 16 GBit/sec
negotiated speed:  16 GBit/sec
OS device name:    /sys/class/scsi_host/host16
```

List all LUNs mapped to host

You can retrieve a list of all LUNs mapped to a host.

```
sanlun lun show -p -v all
```

Show example

```
                    ONTAP Path: vs_sanboot:/vol/sanboot_169/lun
                           LUN: 0
                      LUN Size: 150g
                       Product: cDOT
                   Host Device: 3600a0980383143393124515873683561
              Multipath Policy: service-time 0
                DM-MP Features: 3 queue_if_no_path pg_init_retries 50
              Hardware Handler: 1 alua
            Multipath Provider: Native
----------- --------- ---------- ------- ------------ ----------
----------
            host      vserver            host:
dm-mp       path      path       /dev/   chan:        vserver    major:
state       state     type       node    id:lun       LIF        minor
----------- --------- ---------- ------- ------------ ----------
----------
active      up        primary    sdq    15:0:5:0     lif_18      65:0
active      up        primary    sds    16:0:5:0     lif_17      65:32
active      up        primary    sdac   16:0:7:0     lif_25      65:192
active      up        primary    sdad   15:0:7:0     lif_26      65:208
active      up        secondary  sdt    15:0:4:0     lif_20      65:48
active      up        secondary  sdr    15:0:6:0     lif_19      65:16
active      up        secondary  sdad   16:0:4:0     lif_27      66:96
active      up        secondary  sdan   16:0:6:0     lif_28      66:112
```

List all LUNs mapped to host from a given SVM

You can retrieve a list of all LUNs mapped to a host from a specific storage VM (SVM).

```
sanlun lun show -p -v vs_sanboot
```

Show example

```
                    ONTAP Path: vs_sanboot:/vol/sanboot_169/lun
                           LUN: 0
                      LUN Size: 160g
                       Product: cDOT
                   Host Device: 3600a0980383143393124515873683561
              Multipath Policy: service-time 0
                DM-MP Features: 3 queue_if_no_path pg_init_retries 50
              Hardware Handler: 1 alua
            Multipath Provider: Native
----------- --------- ---------- ------- ------------ ---------------
----------
            host      vserver            host:
dm-mp       path      path       /dev/   chan:        vserver
major:
state       state     type       node    id:lun       LIF
minor
----------- --------- ---------- ------- ------------ ---------------
----------
active      up        primary    sdce    15:0:5:0     lif_16g_5
69:32
active      up        primary    sdfk    16:0:5:0     lif_16g_7
130:96
active      up        primary    sdfm    16:0:7:0     lif_16g_8
130:128
active      up        primary    sdcg    15:0:7:0     lif_16g_6
69:64
active      up        secondary  sdcd    15:0:4:0     lif_16g_1
69:16
active      up        secondary  sdcf    15:0:6:0     lif_16g_2
69:48
active      up        secondary  sdfj    16:0:4:0     lif_16g_3
130:80
active      up        secondary  sdfl    16:0:6:0     lif_16g_4
130:112
```

List all attributes of a given LUN mapped to host

You can retrieve a list of all attributes of a specified LUN mapped to a host.

```
sanlun lun show -p -v vs_sanboot:/vol/sanboot_169/lun
```

Show example

```
                    ONTAP Path: vs_sanboot:/vol/sanboot_169/lun
                           LUN: 0
                      LUN Size: 160g
                       Product: cDOT
                   Host Device: 3600a0980383143393124515873683561
              Multipath Policy: service-time 0
                DM-MP Features: 3 queue_if_no_path pg_init_retries 50
              Hardware Handler: 1 alua
            Multipath Provider: Native
----------- --------- ---------- ------- ------------ -----------------
----------
            host      vserver            host:
dm-mp       path      path       /dev/   chan:        vserver
major:
state       state     type       node    id:lun       LIF
minor
----------- --------- ---------- ------- ------------ -----------------
----------
active      up        primary    sdce    15:0:5:0     lif_16g_5
69:32
active      up        primary    sdfk    16:0:5:0     lif_16g_7
130:96
active      up        primary    sdfm    16:0:7:0     lif_16g_8
130:128
active      up        primary    sdcg    15:0:7:0     lif_16g_6
69:64
active      up        secondary  sdcd    15:0:4:0     lif_16g_1
69:16
active      up        secondary  sdcf    15:0:6:0     lif_16g_2
69:48
active      up        secondary  sdfj    16:0:4:0     lif_16g_3
130:80
active      up        secondary  sdfl    16:0:6:0     lif_16g_4
130:112
```

List the ONTAP SVM identity from which a given LUN is mapped to host

You can retrieve a list of ONTAP SVM identity from which a specific LUN is mapped to a hist.

```
sanlun lun show -m -v vs_sanboot:/vol/sanboot_169/lun
```

Show example

```
                                                                    device
host                  lun
vserver                       lun-pathname                      filename
adapter    protocol   size    product
------------------------------------------------------------------------
---------------------------------------
vs_sanboot                    /vol/sanboot_169/lun              /dev/sdfm
host16     FCP        160g    cDOT
             LUN Serial number: 81C91$QXsh5a
         Controller Model Name: AFF-A400
          Vserver FCP nodename: 2008d039ea1308e5
          Vserver FCP portname: 2010d039ea1308e5
              Vserver LIF name: lif_16g_8
            Vserver IP address: 10.141.12.165
                                10.141.12.161
                                10.141.12.163
           Vserver volume name: sanboot_169
MSID::0x00000000000000000000000809E7CC3
         Vserver snapshot name:
```

List ONTAP LUN attributes by host device filename

You can retrieve a list of ONTAP LUN attributes by a host device filename.

```
sanlun lun show -d /dev/sdce
```

Show example

```
controller(7mode/E-Series)/                          device        host
lun
vserver(cDOT/FlashRay)      lun-pathname           filename
adapter    protocol   size    product
------------------------------------------------------------------------
-----------------------------
vs_sanboot                 /vol/sanboot_169/lun    /dev/sdce     host15
FCP        160g    cDOT
[root@sr630-13-169 ~]#
```

List all SVM target LIF WWPNs attached to host

You can retrieve a list of all SVM target LIF WWPNs attached to a host.

```
sanlun lun show -wwpn
```

Show example

```
controller(7mode/E-Series)/   target
device          host       lun
vserver(cDOT/FlashRay)        wwpn                lun-pathname
filename        adapter    size    product
------------------------------------------------------------------------
-----------------------------------------------
vs_169_16gEmu                 202cd039ea1308e5
/vol/VOL_8g_169_2_8/lun        /dev/sdlo       host18     10g     cDOT
vs_169_16gEmu                 202cd039ea1308e5
/vol/VOL_8g_169_2_9/lun        /dev/sdlp       host18     10g     cDOT
vs_169_16gEmu                 202cd039ea1308e5
/vol/VOL_8g_169_2_7/lun        /dev/sdln       host18     10g     cDOT
vs_169_16gEmu                 202cd039ea1308e5
/vol/VOL_8g_169_2_5/lun        /dev/sdll       host18     10g     cDOT
```

List ONTAP LUNs seen on host by a given SVM target LIF WWPN

You can retrieve a list of ONTAP LUNs noticed on a host by a specified SVM target LIF WWPN.

```
sanlun lun show -wwpn 2010d039ea1308e5
```

Show example

```
controller(7mode/E-Series)/   target
device         host       lun
vserver(cDOT/FlashRay)        wwpn                lun-pathname
filename       adapter    size    product
------------------------------------------------------------------------
----------------------------------------
vs_sanboot                    2010d039ea1308e5   /vol/sanboot_169/lun
/dev/sdfm      host16     160g    cDOT
```

Solaris Host Utilities

Solaris Host Utilities Release Notes

The Solaris Host Utilities release notes describe new features, enhancements, known issues, limitations, and important cautions related to configuring and managing your specific Solaris host with your ONTAP storage system.

For specific information about the operating system versions and updates that the Host Utilities support, see the Interoperability Matrix Tool.

What's new in Solaris Host Utilities 8.0

- Solaris Host Utilities 8.0 introduces the `disksort:false` and `cache-nonvolatile:true` parameter settings. These parameters suppress constant sync calls from the host. The sync calls impact performance and aren't required because ONTAP doesn't have volatile cache. For more information, see Cache Flush Behavior for Flash and NVRAM Storage Devices.
- Solaris Host Utilities 8.0 supports automated updates for FC drivers. The FC drivers that bind the parameters for Solaris 11.4 use SCSI disks. For more information, see the Oracle Support Doc ID 2595926.1.

Known issues and limitations

You should be aware of the following known issues and limitations that might impact performance on your specific host.

Bug ID	Affects version	Title	Description
	Solaris Host Utilities 8.0		The Solaris Host Utilities 8.0 release only supports Solaris 11.4 with SPARC and x86 platforms. For Solaris 11.3 and earlier versions, you need to use the Solaris Host Utilities 6.2.
1385189	Solaris 11.4	Solaris 11.4 FC driver binding changes required in HUK 6.2	Solaris 11.4 and HUK recommendations: FC driver binding is changed from `ssd(4D)` to `sd(4D)`. Move configuration that you have in `ssd.conf` to `sd.conf` as mentioned in Oracle (Doc ID 2595926.1). The behavior varies across newly installed Solaris 11.4 systems and systems upgraded from 11.3 or lower versions.

+
NetApp Bugs Online provides complete information for most known issues, including suggested workarounds where possible. Some keyword combinations and bug types that you might want to use include the following:

- FCP General: Displays FC and host bus adapter (HBA) bugs that are not associated with a specific host.
- FCP - Solaris

What's next

Learn about installing Solaris Host Utilities

Install Solaris Host Utilities

Install Solaris Host Utilities 8.0 for ONTAP storage

The Solaris Host Utilities help you manage ONTAP storage attached to a Solaris host and assist technical support with gathering configuration data.

The Solaris Host Utilities support the following Solaris environments and transport protocols. These are the primary supported environments:

- The native OS with Oracle Solaris I/O Multipathing (MPxIO) and either the FC or iSCSI protocol on a system using either a SPARC or x86/64 processor.
- Veritas Dynamic Multipathing (DMP) with the FC or iSCSI protocol on a system using a SPARC processor.

Solaris Host Utilities 8.0 support the Solaris 11.4 series.

Before you begin

Verify that your iSCSI, FC, or FCoE configuration is supported. You can use the Interoperability Matrix Tool to verify your configuration.

Steps

1. Log in to your host as root.
2. Download a copy of the compressed file containing the Host Utilities from the NetApp Support Site to a directory on your host.

   SPARC CPU

   ```
   netapp_solaris_host_utilities_8_0_sparc.tar.gz
   ```

   x86/x64 CPU

   ```
   netapp_solaris_host_utilities_8_0_amd.tar.gz
   ```

3. Go to the directory on your host containing the download.
4. Unzip the file using the `gunzip` command, and then extract the file using `tar -xvf`:

   ```
   gunzip netapp_solaris_host_utilities_8_0_sparc.tar.gz
   ```

   ```
   tar -xvf netapp_solaris_host_utilities_8_0_sparc.tar
   ```

5. Add the packages that you extracted from the tar file to your host:

   ```
   pkgadd
   ```

The packages are added to the `/opt/NTAP/SANToolkit/bin` directory.

The following example uses the `pkgadd` command to install the Solaris installation package:

```
pkgadd -d ./NTAPSANTool.pkg
```

6. Confirm that the toolkit was successfully installed by using the following command to the installed path:

```
pkgchk
```

Show example output

```
# pkgchk -l -p /opt/NTAP/SANToolkit

Pathname: /opt/NTAP/SANToolkit
Type: directory
Expected mode: 0755
Expected owner: root
Expected group: sys
Referenced by the following packages: NTAPSANTool
Current status: installed

# ls -alR /opt/NTAP/SANToolkit
/opt/NTAP/SANToolkit:
total 1038
drwxr-xr-x   3 root     sys            4 Mar  7 13:11 .
drwxr-xr-x   3 root     sys            3 Mar  7 13:11 ..
drwxr-xr-x   2 root     sys            6 Mar 17 18:32 bin
-r-xr-xr-x   1 root     sys       432666 Dec 31 13:23 NOTICES.PDF

/opt/NTAP/SANToolkit/bin:
total 3350
drwxr-xr-x   2 root     sys            6 Mar 17 18:32 .
drwxr-xr-x   3 root     sys            4 Mar  7 13:11 ..
-r-xr-xr-x   1 root     sys      1297000 Feb  7 22:22 host_config
-r-xr-xr-x   1 root     root         996 Mar 17 18:32 san_version
-r-xr-xr-x   1 root     sys       309700 Feb  7 22:22 sanlun
-r-xr-xr-x   1 root     sys          677 Feb  7 22:22 vidpid.dat

# cd /usr/share/man/man1; ls -al host_config.1 sanlun.1
-r-xr-xr-x   1 root     sys        12266 Feb  7 22:22 host_config.1
-r-xr-xr-x   1 root     sys         9044 Feb  7 22:22 sanlun.1
```

7. Configure the host parameters for your "MPxIO" or "Veritas DMP" environment by using the `/opt/NTAP/SANToolkit/bin/host_config` command with the multipath stack from the command reference:

   ```
   /opt/NTAP/SANToolkit/bin/host_config -setup -protocol fcp|iscsi|mixed
   -multipath mpxio|dmp|non [-noalua] [-mcc 60|90|120]
   ```

For example, if your setup is…	Use the command…
FCP with multipath as MPxIO	`#/opt/NTAP/SANToolkit/bin/host_config -setup -protocol fcp -multipath mpxio` For information on the configuration changes for SnapMirror active sync, see the Knowledge Base article Solaris Host support recommended settings in SnapMirror active sync (formerly SM-BC) configuration.
FCP with multipath as DMP	`#/opt/NTAP/SANToolkit/bin/host_config -setup -protocol fcp -multipath dmp`
FCP on MetroCluster with multipath as MPxIO, and the All Paths Down value is set to 120s. (This is the recommended setting for MetroCluster configurations).	`#/opt/NTAP/SANToolkit/bin/host_config -setup -protocol fcp -multipath mpxio -mcc 120` For more information, see the Knowledge Base article Solaris host support considerations in a MetroCluster configuration.

8. Reboot the host.

   The Host Utilities load the following NetApp recommended timeout parameter settings for ONTAP LUNs.

Show example

```
#prtconf -v |grep NETAPP
   value='NETAPP  LUN' +
   physical-block-size:4096,
   retries-busy:30,
   retries-reset:30,
   retries-notready:300,
   retries-timeout:10,
   throttle-max:64,
   throttle-min:8,
   disksort:false,
   cache-nonvolatile:true'
```

9. Verify the Host Utilities installation:

```
sanlun version
```

What's next?

Learn about the SAN Toolkit.

Install Solaris Host Utilities 6.2 for ONTAP storage

The Solaris Host Utilities help you manage ONTAP storage attached to a Solaris host and assist technical support with gathering configuration data.

The Solaris Host Utilities support several Solaris environments and multiple transport protocols. These are the primary Solaris Host Utilities environments:

- The native OS with MPxIO and either the Fibre Channel (FC) or iSCSI protocol on a system using either a SPARC processor or an x86/64 processor.
- Veritas Dynamic Multipathing (DMP) with either the FC or iSCSI protocol on a system using a SPARC processor, or the iSCSI protocol on a system using an x86/64 processor.

The Solaris Host Utilities 6.2 support the following Solaris series:

- Solaris 11.x
- Solaris 10.x

Before you begin

Verify that your iSCSI, FC, or FCoE configuration is supported. You can use the Interoperability Matrix Tool to verify your configuration.

Steps

1. Log in to your host as root.

2. Download a copy of the compressed file containing the Host Utilities from the NetApp Support Site to a directory on your Solaris host:

SPARC CPU

```
netapp_solaris_host_utilities_6_2_sparc.tar.gz
```

x86/x64 CPU

```
netapp_solaris_host_utilities_6_2_amd.tar.gz
```

3. Go to the directory on your Solaris host containing the download.
4. Unzip the file using the `gunzip` command:

```
gunzip netapp_solaris_host_utilities_6_2_sparc.tar.gz
```

5. Extract the file using the `tar xvf` command:

```
tar xvf netapp_solaris_host_utilities_6_2_sparc.tar
```

6. Add the packages that you extracted from tar file to your host

```
pkgadd
```

The packages are added to the `/opt/NTAP/SANToolkit/bin` directory.

The following example uses the `pkgadd` command to install the Solaris installation package:

```
pkgadd -d ./NTAPSANTool.pkg
```

7. Confirm that the toolkit was successfully installed by using one of the following commands:

```
pkginfo
```

```
ls - al
```

Show example outputs

```
# ls -alR /opt/NTAP/SANToolkit
/opt/NTAP/SANToolkit:
total 1038
drwxr-xr-x   3 root     sys            4 Jul 22  2019 .
drwxr-xr-x   3 root     sys            3 Jul 22  2019 ..
drwxr-xr-x   2 root     sys            6 Jul 22  2019 bin
-r-xr-xr-x   1 root     sys       432666 Sep 13  2017 NOTICES.PDF

/opt/NTAP/SANToolkit/bin:
total 7962
drwxr-xr-x   2 root     sys            6 Jul 22  2019 .
drwxr-xr-x   3 root     sys            4 Jul 22  2019 ..
-r-xr-xr-x   1 root     sys      2308252 Sep 13  2017 host_config
-r-xr-xr-x   1 root     sys          995 Sep 13  2017 san_version
-r-xr-xr-x   1 root     sys      1669204 Sep 13  2017 sanlun
-r-xr-xr-x   1 root     sys          677 Sep 13  2017 vidpid.dat

# (cd /usr/share/man/man1; ls -al host_config.1 sanlun.1)
-r-xr-xr-x   1 root     sys        12266 Sep 13  2017 host_config.1
-r-xr-xr-x   1 root     sys         9044 Sep 13  2017 sanlun.1
```

8. Configure the host parameters for your MPxIO or Veritas DMP environment:

```
/opt/NTAP/SANToolkit/bin/host_config
```

9. Verify the installation:

```
sanlun version
```

What's next?

Learn about the SAN Toolkit.

Learn about the SAN Toolkit for ONTAP storage

Solaris Host Utilities is a NetApp host software that provides a command line toolkit on your Oracle Solaris host. The toolkit is installed when you install the NetApp Host Utilities package. This kit provides the `sanlun` utility which helps you manage LUNs and host bus adapters (HBAs). The `sanlun` command returns information about the LUNs mapped to your host, multipathing, and information necessary to create initiator groups.

The following example output shows the ONTAP LUN information returned for the `sanlun lun show` command:

Show example output

```
#sanlun lun show all
controller(7mode)/ device host lun
vserver(Cmode)                    lun-pathname       filename
adapter protocol size mode
--------------------------------------------------------------------------
------------------------------------------------------------
data_vserver                      /vol/vol1/lun1
/dev/rdsk/c0t600A098038304437522B4E694E49792Dd0s2 qlc3   FCP       10g
cDOT
data_vserver                      /vol/vol0/lun2
/dev/rdsk/c0t600A098038304437522B4E694E497938d0s2 qlc3   FCP       10g
cDOT
data_vserver                      /vol/vol2/lun3
/dev/rdsk/c0t600A098038304437522B4E694E497939d0s2 qlc3   FCP       10g
cDOT
data_vserver                      /vol/vol3/lun4
/dev/rdsk/c0t600A098038304437522B4E694E497941d0s2 qlc3   FCP       10g
cDOT
```

This toolkit is common across all Host Utilities configurations and protocols. As a result, all of the components don't apply to every configuration. Unused components don't affect your system performance.

What's next?

Learn about using the Solaris Host Utilities tool.

Use Solaris Host Utilities commands to verify ONTAP storage configuration

You can use the Solaris Host Utilities sample command reference for an end-to-end validation of the NetApp storage configuration using the Host Utilities tool.

List all host initiators mapped to host

You can retrieve a list of all host initiators mapped to a host.

```
sanlun fcp show adapter -v
```

8.0

Show example for Solaris Host Utilities 8.0

```
adapter name:      qlc0
WWPN:              2100f4e9d40fe3e0
WWNN:              2000f4e9d40fe3e0
driver name:       qlc
model:             7023303
model description: 7101674, Sun Storage 16Gb FC PCIe Universal HBA,
QLogic
serial number:     463916R+1912389772
hardware version:  Not Available
driver version:    230206-5.12
firmware version:  8.08.04
Number of ports:   1 of 2
port type:         Fabric
port state:        Operational
supported speed:   4 GBit/sec, 8 GBit/sec, 16 GBit/sec
negotiated speed:  16 GBit/sec
OS device name:    /dev/cfg/c4

adapter name:      qlc1
WWPN:              2100f4e9d40fe3e1
WWNN:              2000f4e9d40fe3e1
driver name:       qlc
model:             7023303
model description: 7101674, Sun Storage 16Gb FC PCIe Universal HBA,
QLogic
serial number:     463916R+1912389772
hardware version:  Not Available
driver version:    230206-5.12
firmware version:  8.08.04
Number of ports:   2 of 2
port type:         Fabric
port state:        Operational
supported speed:   4 GBit/sec, 8 GBit/sec, 16 GBit/sec
negotiated speed:  16 GBit/sec
OS device name:    /dev/cfg/c5
```

6.2

Show example for Solaris Host Utilities 6.2

```
adapter name:      qlc3
WWPN:              21000024ff17a301
WWNN:              20000024ff17a301
driver name:       qlc
model:             7335902
model description: 7115462, Oracle Storage Dual-Port 32 Gb Fibre
Channel PCIe HBA
serial number:     463916R+1720333838
hardware version:  Not Available
driver version:    210226-5.10
firmware version:  8.08.04
Number of ports:   1 of 2
port type:         Fabric
port state:        Operational
supported speed:   8 GBit/sec, 16 GBit/sec, 32 GBit/sec
negotiated speed:  32 GBit/sec
OS device name:    /dev/cfg/c7

adapter name:      qlc2
WWPN:              21000024ff17a300
WWNN:              20000024ff17a300
driver name:       qlc
model:             7335902
model description: 7115462, Oracle Storage Dual-Port 32 Gb Fibre
Channel PCIe HBA
serial number:     463916R+1720333838
hardware version:  Not Available
driver version:    210226-5.10
firmware version:  8.08.04
Number of ports:   2 of 2
port type:         Fabric
port state:        Operational
supported speed:   8 GBit/sec, 16 GBit/sec, 32 GBit/sec
negotiated speed:  16 GBit/sec
OS device name:    /dev/cfg/c6
```

List all LUNs mapped to host

You can retrieve a list of all LUNs mapped to a host.

```
sanlun lun show -p -v all
```

8.0

Show example for Solaris Host Utilities 8.0

```
ONTAP Path: sanboot_unix:/vol/test1/lun1
       LUN: 0
       LUN Size: 21g
       Host Device:
/dev/rdsk/c0t600A098038314B314E5D574632365A51d0s2
       Mode: C
       Multipath Provider: Sun Microsystems
       Multipath Policy: Native
```

6.2

Show example for Solaris Host Utilities 6.2

```
                    ONTAP Path: data_vserver:/vol1/lun1
                           LUN: 1
                      LUN Size: 10g
                   Host Device:
/dev/rdsk/c0t600A0980383044485A3F4E694E4F775Ad0s2
                          Mode: C
             Multipath Provider: Sun Microsystems
               Multipath Policy: Native
```

List all LUNs mapped to host from a given SVM/ List all attributes of a given LUN mapped to host

You can retrieve a list of all LUNs mapped to a host from a specific SVM.

```
sanlun lun show -p -v <svm_name>
```

8.0

Show example for Solaris Host Utilities 8.0

```
ONTAP Path: sanboot_unix:/vol/test1/lun1
       LUN: 0
       LUN Size: 20g
       Host Device:
/dev/rdsk/c0t600A098038314B314E5D574632365A51d0s2
       Mode: C
       Multipath Provider: Sun Microsystems
       Multipath Policy: Native
```

6.2

Show example for Solaris Host Utilities 6.2

```
ONTAP Path: sanboot_unix:/vol/sol_boot/sanboot_lun
                           LUN: 0
                      LUN Size: 180.0g
```

List ONTAP LUN attributes by host device filename

You can retrieve a list of all ONTAP LUN attributes by specifying a host device filename.

```
sanlun lun show all
```

8.0

Show example for Solaris Host Utilities 8.0

```
controller(7mode/E-Series)/
device
vserver(cDOT/FlashRay)       lun-pathname
filename
------------------------------------------------------------------------
-----------------------------------------
sanboot_unix                /vol/test1/lun1
/dev/rdsk/
c0t600A098038314B314E5D574632365A51d0s2

host adapter    protocol lun size   product
-----------------------------------------
qlc1            FCP      20g        cDOT
```

6.2

Show example for Solaris Host Utilities 6.2

```
controller(7mode/E-Series)/
device
vserver(cDOT/FlashRay)       lun-pathname
filename
------------------------------------------------------------------------
-----------------------------------------
sanboot_unix                 /vol/sol_193_boot/chatsol_193_sanboot
/dev/rdsk/c0t600A098038304437522B4E694E4A3043d0s2

host adapter    protocol lun size   product
-----------------------------------------
qlc3            FCP      180.0g     cDOT
```

Windows Host Utilities

Windows Host Utilities Release Notes

The release notes describe new features, enhancements, known issues, limitations, and important cautions related to configuring and managing your specific Windows host with your ONTAP storage system.

For specific information about the operating system versions and updates that the Host Utilities support, see the Interoperability Matrix Tool.

What's new in Windows Host Utilities 8.0

Windows Host Utilities 8.0 include additional parameter settings for NVMe/FC drivers. These parameters are automatically loaded when you install the Windows Host Utilities 8.0.

What's new in Windows Host Utilities 7.2

Windows Host Utilities 7.2 includes support for NVMe/FC driver parameter settings and bug fixes for new vendor cards.

What's new in Windows Host Utilities 7.1

Windows Host Utilities 7.1 includes support for E-Series storage systems, ONTAP 9 software, and enhanced iSCSI timeouts for faster failover.

Known issues and limitations

You should be aware of the following known issues and limitations that might impact performance on your specific host.

Title	Affects version	Description	Workaround
Running the `linux_gos_timeout-install.sh` script is no longer required on Hyper-V guests running Red Hat Enterprise Linux or SUSE Linux Enterprise Server	7.1	You are no longer required to run the `linux_gos_timeout-install.sh` script to change disk timeouts on Red Hat Enterprise Linux 5, Red Hat Enterprise Linux 6, or SUSE Linux Enterprise Server 11 Hyper-V guests because the default timeout settings are being used. The `linux_gos_timeout-install.sh` script that was included in previous versions of the Host Utilities is no longer included in the Windows Host Utilities 7.1 version. Windows Host Utilities 7.1 uses the default disk timeout settings.	Not Applicable

NetApp Bugs Online provides complete information for most known issues, including suggested workarounds where possible.

What's next

Learn about installing Windows Host Utilities

Install Windows Host Utilities

Install Windows Host Utilities 8.0 for ONTAP storage

The Windows Host Utilities enable you to connect a Windows host computer to NetApp storage systems.

The Windows Host Utilities support the following versions of Windows:

- Windows 2025
- Windows 2022
- Windows 2019
- Windows 2016

Windows Host Utilities includes an installation program that sets the required Windows registry and Host Bus Adapter (HBA) parameters so that a Windows host can correctly handle the storage system behaviors for ONTAP platforms.

When you install the Host Utilities software, the installer sets the required Windows registry and HBA parameters.

The following programs and files are installed on the Windows host computer. The default directory is `C:\Program Files\NetApp\Windows Host Utilities`.

Program	Purpose
`\NetAppQCLI\fcconfig.exe`	Used by the installation program to set the HBA parameters.
`\NetAppQCLI\fcconfig.ini`	Used by the installation program to set the HBA parameters.
`san_version.exe`	Displays the version of the Host Utilities and FC HBAs.

The Host Utilities support different Windows host configurations, protocols, and multipathing options. For more information, see the Interoperability Matrix Tool.

Step 1: Verify your host and storage system configuration

Verify that your host and storage system configuration are supported before you install Windows Host Utilities for the supported Windows version.

Steps

1. Check the supported configuration in the Interoperability Matrix Tool.
2. Check the hotfixes required for your Windows Server host version in the SAN host Windows documentation.

   For example, the Configure Windows Server 2025 for ONTAP storage documentation provides the instructions for installing Windows hotfixes for Windows Server 2025.

3. Add the iSCSI or FCP license.

4. Start the target service.
5. Verify your cabling.

   Refer to the SAN configuration reference for detailed cabling and configuration information.

Step 2: Configure FC HBAs and switches

Install and configure one or more supported FC host bus adapters (HBAs) for FC connections to the storage system.

The Windows Host Utilities installer sets the required FC HBA settings.

> You should only allow the Windows Host Utilities installer to set the HBA parameters. This ensures that the Windows host can correctly handle the storage system behaviors for ONTAP platforms.

Steps

1. Install one or more supported FC HBAs according to the instructions provided by the HBA vendor.
2. Obtain the supported HBA drivers and management utilities and install them according to the instructions provided by the HBA vendor.
3. Connect the HBAs to your FC switches or directly to the storage system.
4. Create zones on the FC switch according to your FC switch documentation.
5. For ONTAP, zone the switch by the WWPN. Be sure to use the WWPN of the logical interfaces (LIFs) and not the WWPN of the physical ports on the storage controllers. Refer to the SAN configuration reference for more information.

Step 3: Install the Windows Host Utilities

The installation program installs the Windows Host Utilities package and sets the Windows registry and HBA settings.

About this task

Optionally, you can include multipathing support when you install the Windows Host Utilities software package. The installer prompts you for the following options:

- Choose `MPIO` if you have more than one path from the Windows host or virtual machine to the storage system.
- Choose `no MPIO` only if you're using a single path to the storage system.

You can also choose multipathing support using a Windows command line prompt.

For Hyper-V guests, raw (pass-through) disks don't appear in the guest OS if you choose multipathing support. You can use either raw disks or MPIO, but you can't use both in the guest OS.

> If you don't install the MPIO software, the Windows OS might see each path as a separate disk. This can lead to data corruption.

> Windows XP or Windows Vista running in a Hyper-V virtual machine doesn't support MPIO.

Steps

You can install the Host Utilities interactively or by using the Windows command line. The new Host Utilities installation package must be in a path that is accessible to the Windows host.

Install interactively

Install the Windows Host Utilities software package interactively by running the installation program and following the prompts.

1. Download the executable file from the NetApp Support Site.
2. Change to the directory where you downloaded the executable file.
3. Run the `netapp_windows_host_utilities_8.0.0_x64` file and follow the instructions on the screen.
4. Reboot the Windows host when prompted.

Install noninteractively

Perform a noninteractive installation of the Host Utilities by using the Windows command line. The system automatically reboots when the installation is complete.

1. Enter the following command at the Windows command prompt:

```
msiexec /i installer.msi /quiet MULTIPATHING= {0 | 1}
[INSTALLDIR=inst_path]
```

   - `installer` is the name of the `.msi` file for your CPU architecture.
   - MULTIPATHING specifies whether MPIO support is installed. The allowed values are "0" for no and "1" for yes.
   - `inst_path` is the path where the Host Utilities files are installed. The default path is `C:\Program Files\NetApp\Windows Host Utilities\`.

> To see the standard Microsoft Installer (MSI) options for logging and other functions, enter `msiexec /help` at the Windows command prompt. For example, the `msiexec /i install.msi /quiet /l*v <install.log> LOGVERBOSE=1` command displays logging information.

What's next?

Learn about the Windows Host Utilities configuration for ONTAP storage

Install Windows Host Utilities 7.2 for ONTAP storage

The Windows Host Utilities enable you to connect a Windows host computer to NetApp storage systems.

The Windows Host Utilities support the following versions of Windows:

- Windows 2025
- Windows 2022

- Windows 2019
- Windows 2016
- Windows 2012R2
- Windows 2012

Windows Host Utilities includes an installation program that sets the required Windows registry and Host Bus Adapter (HBA) parameters so that a Windows host can correctly handle the storage system behaviors for ONTAP and E-Series platforms.

When you install the Host Utilities software, the installer sets the required Windows registry and HBA parameters.

The following programs and files are installed on the Windows host computer. The default directory is `C:\Program Files\NetApp\Windows Host Utilities`.

Program	Purpose
`emulexhba.reg`	Troubleshooting program; run this program only if instructed to do so by technical support personnel.
`\NetAppQCLI\fcconfig.exe`	Used by the installation program to set the HBA parameters.
`\NetAppQCLI\fcconfig.ini`	Used by the installation program to set the HBA parameters.
`\NetAppQCLI*.*`	Used by the installation program to set the QLogic FC HBA parameters.
`san_version.exe`	Displays the version of the Host Utilities and FC HBAs.

The Host Utilities support different Windows host configurations, protocols, and multipathing options. For more information, see the Interoperability Matrix Tool.

Step 1: Verify your host and storage system configuration

Verify that your host and storage system configuration are supported before you install Windows Host Utilities for the supported Windows version.

Steps

1. Check the supported configuration in the Interoperability Matrix Tool.
2. Check the hotfixes required for your Windows Server host version in the SAN host Windows documentation.

   For example, the Configure Windows Server 2022 for ONTAP storage documentation provides the instructions for installing Windows hotfixes for Windows server 2022.

3. Add the iSCSI or FCP license.

   The FC and iSCSI protocols don't require licenses on E-Series storage systems using the SANtricity Storage Manager.

4. Start the target service.

5. Verify your cabling.

   Refer to the SAN configuration reference for detailed cabling and configuration information.

Step 2: Configure FC HBAs and switches

Install and configure one or more supported FC host bus adapters (HBAs) for FC connections to the storage system.

The Windows Host Utilities installer sets the required FC HBA settings.

> You should only allow the Windows Host Utilities installer to set the HBA parameters. This ensures that the Windows host can correctly handle the storage system behaviors for ONTAP platforms.

Steps

1. Install one or more supported FC HBAs according to the instructions provided by the HBA vendor.
2. Obtain the supported HBA drivers and management utilities and install them according to the instructions provided by the HBA vendor.
3. Connect the HBAs to your FC switches or directly to the storage system.
4. Create zones on the FC switch according to your FC switch documentation.
5. For ONTAP, zone the switch by the WWPN. Be sure to use the WWPN of the logical interfaces (LIFs) and not the WWPN of the physical ports on the storage controllers. Refer to the SAN configuration reference for more information.

Step 3: Install the Windows Host Utilities

The installation program installs the Windows Host Utilities package and sets the Windows registry and HBA settings.

About this task

Optionally, you can include multipathing support when you install the Windows Host Utilities software package. The installer prompts you for the following options:

- Choose `MPIO` if you have more than one path from the Windows host or virtual machine to the storage system.
- Choose `no MPIO` only if you're using a single path to the storage system.

You can also choose multipathing support using a Windows command line prompt.

For Hyper-V guests, raw (pass-through) disks don't appear in the guest OS if you choose multipathing support. You can use either raw disks or MPIO, but you can't use both in the guest OS.

> If you don't install the MPIO software, the Windows OS might see each path as a separate disk. This can lead to data corruption.

> Windows XP or Windows Vista running in a Hyper-V virtual machine doesn't support MPIO.

Steps

You can install the Host Utilities interactively or by using the Windows command line. The new Host Utilities

installation package must be in a path that is accessible to the Windows host.

Install interactively

Install the Host Utilities software package interactively by running the Host Utilities installation program and following the prompts.

1. Download the executable file from the NetApp Support Site.
2. Change to the directory where you downloaded the executable file.
3. Run the `netapp_windows_host_utilities_7.2_x64` file and follow the instructions on the screen.
4. Reboot the Windows host when prompted.

Install noninteractively

Perform a noninteractive installation of the Host Utilities by using the Windows command line. The system automatically reboots when the installation is complete.

1. Enter the following command at the Windows command prompt:

```
msiexec /i installer.msi /quiet MULTIPATHING= {0 | 1}
[INSTALLDIR=inst_path]
```

   - `installer` is the name of the `.msi` file for your CPU architecture.
   - MULTIPATHING specifies whether MPIO support is installed. The allowed values are "0" for no and "1" for yes.
   - `inst_path` is the path where the Host Utilities files are installed. The default path is `C:\Program Files\NetApp\Windows Host Utilities\`.

> To see the standard Microsoft Installer (MSI) options for logging and other functions, enter `msiexec /help` at the Windows command prompt. For example, the `msiexec /i install.msi /quiet /l*v <install.log> LOGVERBOSE=1` command displays logging information.

What's next?

Learn about the Windows Host Utilities configuration for ONTAP storage

Install Windows Host Utilities 7.1 for ONTAP storage

The Windows Host Utilities enable you to connect a Windows host computer to NetApp storage systems.

Windows Host Utilities supports the following versions of Windows:

- Windows 2022
- Windows 2019
- Windows 2016

- Windows 2012R2
- Windows 2012

Windows Host Utilities includes an installation program that sets the required Windows registry and Host Bus Adapter (HBA) parameters so that a Windows host can correctly handle the storage system behaviors for ONTAP and E-Series platforms.

When you install the Host Utilities software, the installer sets the required Windows registry and HBA parameters.

The following programs and files are installed on the Windows host computer. The default directory is `C:\Program Files\NetApp\Windows Host Utilities`.

Program	Purpose
`emulexhba.reg`	Troubleshooting program; run this program only if instructed to do so by technical support personnel.
`\NetAppQCLI\fcconfig.exe`	Used by the installation program to set HBA parameters.
`\NetAppQCLI\fcconfig.ini`	Used by the installation program to set HBA parameters.
`\NetAppQCLI*.*`	Used by the installation program to set QLogic FC HBA parameters.
`san_version.exe`	Displays the version of the Host Utilities and FC HBAs.

The Host Utilities support different Windows host configurations, protocols, and multipathing options. See the Interoperability Matrix Tool for the current list of supported configurations.

Step 1: Verify your host and storage system configurations

Verify that your host and storage system configuration are supported before you install Windows Host Utilities for the supported Windows version.

Steps

1. Check the supported configuration in the Interoperability Matrix Tool.
2. Check the hotfixes required for your Windows Server host version in the SAN host Windows documentation.

   For example, the Configure Windows Server 2022 for ONTAP storage documentation provides the instructions for installing Windows hotfixes for Windows server 2022.

3. Add the iSCSI or FCP license.

   The FC and iSCSI protocols don't require licenses on E-Series storage systems using the SANtricity Storage Manager.

4. Start the target service.
5. Verify your cabling.

Refer to the SAN configuration reference for detailed cabling and configuration information.

Step 2: Configure FC HBAs and switches

Install and configure one or more supported FC host bus adapters (HBAs) for FC connections to the storage system.

The Windows Host Utilities installer sets the required FC HBA settings.

> You should only allow the Windows Host Utilities installer to set the HBA parameters. This ensures that the Windows host can correctly handle the storage system behaviors for ONTAP platforms.

Steps

1. Install one or more supported FC HBAs according to the instructions provided by the HBA vendor.
2. Obtain the supported HBA drivers and management utilities and install them according to the instructions provided by the HBA vendor.
3. Connect the HBAs to your FC switches or directly to the storage system.
4. Create zones on the FC switch according to your FC switch documentation.
5. For ONTAP, zone the switch by the WWPN. Be sure to use the WWPN of the logical interfaces (LIFs) and not the WWPN of the physical ports on the storage controllers. Refer to the SAN configuration reference for more information.

Step 3: Install the Windows Host Utilities

The installation program installs the Windows Host Utilities package and sets the Windows registry and HBA settings.

About this task

Optionally, you can include multipathing support when you install the Windows Host Utilities software package. The installer prompts you for the following options:

- Choose `MPIO` if you have more than one path from the Windows host or virtual machine to the storage system.
- Choose `no MPIO` only if you're using a single path to the storage system.

You can also choose multipathing support using a Windows command line prompt.

For Hyper-V guests, raw (pass-through) disks don't appear in the guest OS if you choose multipathing support. You can use either raw disks or MPIO, but you can't use both in the guest OS.

> If you don't install the MPIO software, the Windows OS might see each path as a separate disk. This can lead to data corruption.

> Windows XP or Windows Vista running in a Hyper-V virtual machine doesn't support MPIO.

Steps

You can install the Host Utilities interactively or by using the Windows command line. The new Host Utilities installation package must be in a path that is accessible to the Windows host.

Install interactively

Install the Host Utilities software package interactively by running the Host Utilities installation program and following the prompts.

1. Download the executable file from the NetApp Support Site.
2. Change to the directory from which you downloaded the executable file.
3. Run the `netapp_windows_host_utilities_7.1_x64` file and follow the instructions on the screen.
4. Reboot the Windows host when prompted.

Install noninteractively

Perform a noninteractive installation of the Host Utilities by using the Windows command line. The system automatically reboots when the installation is complete.

1. Enter the following command at a Windows command prompt:

```
msiexec /i installer.msi /quiet MULTIPATHING= {0 | 1}
[INSTALLDIR=inst_path]
```

   - `installer` is the name of the `.msi` file for your CPU architecture
   - MULTIPATHING specifies whether MPIO support is installed. Allowed values are "0" for no, "1" for yes
   - `inst_path` is the path where the Host Utilities files are installed. The default path is `C:\Program Files\NetApp\Windows Host Utilities\`.

> To see the standard Microsoft Installer (MSI) options for logging and other functions, enter `msiexec /help` at a Windows command prompt. For example, the`msiexec /i install.msi /quiet /l*v <install.log> LOGVERBOSE=1` command displays logging information.

What's next?

Learn about the Windows Host Utilities configuration for ONTAP storage

Review the Windows Host Utilities configuration for ONTAP storage

The Windows Host Utilities require certain registry and parameter settings so that a Windows host correctly handles the ONTAP storage system behavior.

These registry and parameter values affect how a Windows host responds to a delay or loss of data, for example, correctly handling events such as a storage failover.

Some of the registry and parameter values don't apply to the device-specific module (DSM) for SANtricity Storage Manager. Overlap between values set by the Windows Host Utilities and the DSM for SANtricity Storage Manager don't cause conflicts.

A Windows host also requires certain FC, NVMe/FC, and iSCSI HBA parameters to successfully handle storage system events:

- Beginning with Windows Host Utilities 7.2, the installer sets the Windows, iSCSI, FC, and NVMe/FC parameters to the supported values.
- Beginning with Windows Host Utilities 7.1, the installer sets the Windows, FC, and NVMe/FC HBA parameters to the supported values. You need to manually set the iSCSI HBA parameters.

The installer sets different values when you specify Microsoft Multipath I/O (MPIO) support during the installation.

> You need to change the LimTransferSize parameter value after installing Windows Host Utilities 7.2. See NVME parameters.
>
> For all other parameters and registry keys for Windows Host Utilities 8.0, 7.2, or 7.1, you don't need to change the values unless technical support advises you to do so.

Review the configuration settings for Windows Host Utilities 8.0

The Windows Host Utilities installer automatically sets the Windows registry values and the iSCSI and NVMe HBA parameters based on the choices that you make during the installation. Review these values and the operating system version.

Windows registry values

All Windows registry values are in decimal unless otherwise stated.

HKLM is the abbreviation for `HKEY_LOCAL_MACHINE`.

Registry key	Value	Set registry key…
HKLM\SYSTEM\CurrentControlSet\Services\msdsm\Parameters\DsmMaximumRetryTimeDuringStateTransition	120	When MPIO support is specified and your server is Windows Server 2025, 2022, 2019, or 2016
HKLM\SYSTEM\CurrentControlSet\Services\msdsm\Parameters\DsmMaximumStateTransitionTime	120	When MPIO support is specified and your server is Windows Server 2025, 2022, 2019, or 2016
HKLM\SYSTEM\CurrentControlSet\Services\msdsm\Parameters\DsmSupportedDeviceList	"NETAPP LUN", "NETAPP LUN C-Mode" "NVMe NetApp ONTAP Con"	When MPIO support is specified
HKLM\SYSTEM\CurrentControlSet\Services\ClusDisk\Parameters\ManageDisksOnSystemBuses	1	Always
HKLM\SYSTEM\CurrentControlSet\Control\Class\{iSCSI_driver_GUID}\instance_ID\Parameters\MaxRequestHoldTime	30	Always
HKLM\SYSTEM\CurrentControlSet\Control\MPDEV\MPIOSupportedDeviceList	"NETAPP LUN", "NETAPP LUN C-Mode", "NVMe NetApp ONTAP Con"	When MPIO support is specified
HKLM\SYSTEM\CurrentControlSet\Services\mpio\Parameters\PathRecoveryInterval	30	When your server is Windows Server 2025, 2022, 2019, or 2016
HKLM\SYSTEM\CurrentControlSet\Services\mpio\Parameters\PathVerifyEnabled	1	When MPIO support is specified

Registry key	Value	Set registry key…
HKLM\SYSTEM\CurrentControlSet\Services\msdsm\Parameters\PathVerifyEnabled	1	When MPIO support is specified and your server is Windows Server 2025, 2022, 2019, or 2016
HKLM\SYSTEM\CurrentControlSet\Services\vnetapp\Parameters\PathVerifyEnabled	0	When MPIO support is specified
HKLM\SYSTEM\CurrentControlSet\Services\mpio\Parameters\PDORemovePeriod	130	When MPIO support is specified
HKLM\SYSTEM\CurrentControlSet\Services\msdsm\Parameters\PDORemovePeriod	130	When MPIO support is specified and your server is Windows Server 2025, 2022, 2019, or 2016
HKLM\SYSTEM\CurrentControlSet\Services\vnetapp\Parameters\PDORemovePeriod	130	When MPIO support is specified
HKLM\SYSTEM\CurrentControlSet\Services\mpio\Parameters\RetryCount	6	When MPIO support is specified
HKLM\SYSTEM\CurrentControlSet\Services\msdsm\Parameters\RetryCount	6	When MPIO support is specified and your server is Windows Server 2025, 2022, 2019, or 2016
HKLM\SYSTEM\CurrentControlSet\Services\mpio\Parameters\RetryInterval	1	When MPIO support is specified
HKLM\SYSTEM\CurrentControlSet\Services\msdsm\Parameters\RetryInterval	1	When MPIO support is specified and your server is Windows Server 2025, 2022, 2019, or 2016
HKLM\SYSTEM\CurrentControlSet\Services\vnetapp\Parameters\RetryInterval	1	When MPIO support is specified
HKLM\SYSTEM\CurrentControlSet\Services\disk\TimeOutValue	120	When no MPIO support is selected
	60	When MPIO support is specified
HKLM\SYSTEM\CurrentControlSet\Services\mpio\Parameters\UseCustomPathRecoveryInterval	1	When MPIO support is specified and your server is Windows Server 2025, 2022, 2019, or 2016

iSCSI HBA values

All iSCSI HBA values are in decimal unless otherwise stated.

HKLM is the abbreviation for `HKEY_LOCAL_MACHINE`.

Registry key	Value	Set registry key…
HKLM\SYSTEM\CurrentControlSet\Control\Class\{iSCSI_driver_GUID}\instance_ID\Parameters\IPSecConfigTimeout	60	Always
HKLM\SYSTEM\CurrentControlSet\Control\Class\{iSCSI_driver_GUID}\instance_ID\Parameters \LinkDownTime	10	Always
HKLM\SYSTEM\CurrentControlSet\Control\Class\{iSCSI_driver_GUID}\instance_ID\Parameters\MaxRequestHoldTime	120	When no MPIO support is selected

NVMe parameters

Windows Host Utilities 8.0 updates the following NVMe Emulex driver parameters during installation:

- EnableNVMe = 1
- NVMEMode = 0

Review the configuration settings for Windows Host Utilities 7.2

The Windows Host Utilities installer automatically sets the Windows registry values and the iSCSI and NVMe HBA parameters based on the choices that you make during the installation. Review these values and the operating system version.

Windows registry values

All Windows registry values are in decimal unless otherwise stated.

HKLM is the abbreviation for `HKEY_LOCAL_MACHINE`.

Registry key	Value	Set registry key…
HKLM\SYSTEM\CurrentControlSet\Services\msdsm\Parameters\DsmMaximumRetryTimeDuringStateTransition	120	When MPIO support is specified and your server is Windows Server 2025, 2022, 2019, 2016, or 2012 R2
HKLM\SYSTEM\CurrentControlSet\Services\msdsm\Parameters\DsmMaximumStateTransitionTime	120	When MPIO support is specified and your server is Windows Server 2025, 2022, 2019, 2016, or 2012 R2
HKLM\SYSTEM\CurrentControlSet\Services\msdsm\Parameters\DsmSupportedDeviceList	"NETAPP LUN", "NETAPP LUN C-Mode" "NVMe NetApp ONTAP Con"	When MPIO support is specified
HKLM\SYSTEM\CurrentControlSet\Services\ClusDisk\Parameters\ManageDisksOnSystemBuses	1	Always
HKLM\SYSTEM\CurrentControlSet\Control\Class\{iSCSI_driver_GUID}\instance_ID\Parameters\MaxRequestHoldTime	30	Always
HKLM\SYSTEM\CurrentControlSet\Control\MPDEV\MPIOSupportedDeviceList	"NETAPP LUN", "NETAPP LUN C-Mode", "NVMe NetApp ONTAP Con"	When MPIO support is specified
HKLM\SYSTEM\CurrentControlSet\Services\mpio\Parameters\PathRecoveryInterval	30	When your server is Windows Server 2025, 2022, 2019, 2016, or 2012 R2
HKLM\SYSTEM\CurrentControlSet\Services\mpio\Parameters\PathVerifyEnabled	1	When MPIO support is specified

Registry key	Value	Set registry key…
HKLM\SYSTEM\CurrentControlSet\Services\msdsm\Parameters\PathVerifyEnabled	1	When MPIO support is specified and your server is Windows Server 2025, 2022, 2019, 2016, or 2012 R2
HKLM\SYSTEM\CurrentControlSet\Services\vnetapp\Parameters\PathVerifyEnabled	0	When MPIO support is specified
HKLM\SYSTEM\CurrentControlSet\Services\mpio\Parameters\PDORemovePeriod	130	When MPIO support is specified
HKLM\SYSTEM\CurrentControlSet\Services\msdsm\Parameters\PDORemovePeriod	130	When MPIO support is specified and your server is Windows Server 2025, 2022, 2019, 2016, or 2012 R2
HKLM\SYSTEM\CurrentControlSet\Services\vnetapp\Parameters\PDORemovePeriod	130	When MPIO support is specified
HKLM\SYSTEM\CurrentControlSet\Services\mpio\Parameters\RetryCount	6	When MPIO support is specified
HKLM\SYSTEM\CurrentControlSet\Services\msdsm\Parameters\RetryCount	6	When MPIO support is specified and your server is Windows Server 2025, 2022, 2019, 2016, or 2012 R2
HKLM\SYSTEM\CurrentControlSet\Services\mpio\Parameters\RetryInterval	1	When MPIO support is specified
HKLM\SYSTEM\CurrentControlSet\Services\msdsm\Parameters\RetryInterval	1	When MPIO support is specified and your server is Windows Server 2025, 2022, 2019, 2016, or 2012 R2
HKLM\SYSTEM\CurrentControlSet\Services\vnetapp\Parameters\RetryInterval	1	When MPIO support is specified
HKLM\SYSTEM\CurrentControlSet\Services\disk\TimeOutValue	120	When no MPIO support is selected
	60	When MPIO support is specified
HKLM\SYSTEM\CurrentControlSet\Services\mpio\Parameters\UseCustomPathRecoveryInterval	1	When MPIO support is specified and your server is Windows Server 2025, 2022, 2019, 2016, or 2012 R2

iSCSI HBA values

All iSCSI HBA values are in decimal unless otherwise stated.

HKLM is the abbreviation for `HKEY_LOCAL_MACHINE`.

Registry key	Value	Set registry key…
HKLM\SYSTEM\CurrentControlSet\Control\Class\{iSCSI_driver_GUID}\instance_ID\Parameters\IPSecConfigTimeout	60	Always
HKLM\SYSTEM\CurrentControlSet\Control\Class\{iSCSI_driver_GUID}\instance_ID\Parameters \LinkDownTime	10	Always
HKLM\SYSTEM\CurrentControlSet\Control\Class\{iSCSI_driver_GUID}\instance_ID\Parameters\MaxRequestHoldTime	120	When no MPIO support is selected

NVMe parameters

Windows Host Utilities 7.2 updates the following NVMe Emulex driver parameters during installation:

- EnableNVMe = 1
- NVMEMode = 0
- LimTransferSize=1

  The LimTransferSize parameter is automatically set to "1" when you install Windows Host Utilities 7.2. After installation, you change the LimTransferSize value to "0" and reboot the server.

Review the configuration settings for Windows Host Utilities 7.1

The Windows Host Utilities installer automatically sets registry values based on the choices that you make during installation. Review these registry values and the operating system version. All values are in decimal unless otherwise noted.

`HKLM` is the abbreviation for `HKEY_LOCAL_MACHINE`.

Registry key	Value	Set registry key…
HKLM\SYSTEM\CurrentControlSet\Services\msdsm\Parameters\DsmMaximumRetryTimeDuringStateTransition	120	When MPIO support is specified and your server is Windows Server 2016, 2012 R2, 2012, 2008 R2, or 2008, except if Data ONTAP DSM is detected

Registry key	Value	Set registry key…
HKLM\SYSTEM\CurrentControlSet\Services \msdsm\Parameters \DsmMaximumStateTransitionTime	120	When MPIO support is specified and your server is Windows Server 2016, 2012 R2, 2012, 2008 R2, or 2008, except if Data ONTAP DSM is detected
HKLM\SYSTEM\CurrentControlSet\Services\msdsm \Parameters\DsmSupportedDeviceList	"NETAPP LUN"	When MPIO support is specified
	"NETAPP LUN", "NETAPP LUN C-Mode"	When MPIO support is specified, except if Data ONTAP DSM is detected
HKLM\SYSTEM\CurrentControlSet\Control\Class {iSCSI_driver_GUID}\ instance_ID\Parameters \IPSecConfigTimeout	60	Always, except when Data ONTAP DSM is detected
HKLM\SYSTEM\CurrentControlSet\Control \Class{iSCSI_driver_GUID} \ instance_ID\Parameters\LinkDownTime	10	Always
HKLM\SYSTEM\CurrentControlSet\Services\ClusDisk \Parameters\ManageDisksOnSystemBuses	1	Always, except when Data ONTAP DSM is detected
HKLM\SYSTEM\CurrentControlSet\Control \Class{iSCSI_driver_GUID} \ instance_ID\Parameters\MaxRequestHoldTime	120	When no MPIO support is selected
	30	Always, except when Data ONTAP DSM is detected
HKLM\SYSTEM\CurrentControlSet \Control\MPDEV\MPIOSupportedDeviceList	"NETAPP LUN"	When MPIO support is specified
	"NETAPP LUN", "NETAPP LUN C-Mode"	When MPIO is support-specified, except if Data ONTAP DSM is detected
HKLM\SYSTEM\CurrentControlSet\Services\mpio \Parameters\PathRecoveryInterval	40	When your server is Windows Server 2008, Windows Server 2008 R2, Windows Server 2012, Windows Server 2012 R2, or Windows Server 2016 only
HKLM\SYSTEM\CurrentControlSet\Services\mpio \Parameters\PathVerifyEnabled	0	When MPIO support is specified, except if Data ONTAP DSM is detected
HKLM\SYSTEM\CurrentControlSet\Services\msdsm \Parameters\PathVerifyEnabled	0	When MPIO support is specified, except if Data ONTAP DSM is detected

Registry key	Value	Set registry key…
HKLM\SYSTEM\CurrentControlSet\Services \msdsm\Parameters\PathVerifyEnabled	0	When MPIO support is specified and your server is Windows Server 2016, 2012 R2, 2012, 2008 R2, or 2008, except if Data ONTAP DSM is detected
HKLM\SYSTEM\CurrentControlSet\Services \msiscdsm\Parameters\PathVerifyEnabled	0	When MPIO support is specified and your server is Windows Server 2003, except if Data ONTAP DSM is detected
HKLM\SYSTEM\CurrentControlSet\Services\vnetapp \Parameters\PathVerifyEnabled	0	When MPIO support is specified, except if Data ONTAP DSM is detected
HKLM\SYSTEM\CurrentControlSet\Services\mpio \Parameters\PDORemovePeriod	130	When MPIO support is specified, except if Data ONTAP DSM is detected
HKLM\SYSTEM\CurrentControlSet\Services\msdsm \Parameters\PDORemovePeriod	130	When MPIO support is specified and your server is Windows Server 2016, 2012 R2, 2012, 2008 R2, or 2008, except if Data ONTAP DSM is detected
HKLM\SYSTEM\CurrentControlSet\Services\msiscdsm \Parameters\PDORemovePeriod	130	When MPIO support is specified and your server is Windows Server 2003, except if Data ONTAP DSM is detected
HKLM\SYSTEM\CurrentControlSet\Services \vnetapp \Parameters\PDORemovePeriod	130	When MPIO support is specified, except if Data ONTAP DSM is detected
HKLM\SYSTEM\CurrentControlSet\Services \mpio\Parameters\RetryCount	6	When MPIO support is specified, except if Data ONTAP DSM is detected
HKLM\SYSTEM\CurrentControlSet\Services\msdsm \Parameters\RetryCount	6	When MPIO support is specified and your server is Windows Server 2016, 2012 R2, 2012, 2008 R2, or 2008, except if Data ONTAP DSM is detected
HKLM\SYSTEM\CurrentControlSet\Services \msiscdsm\Parameters\RetryCount	6	When MPIO support is specified and your server is Windows Server 2003, except if Data ONTAP DSM is detected
HKLM\SYSTEM\CurrentControlSet\Services \vnetapp\Parameters\RetryCount	6	When MPIO support is specified, except if Data ONTAP DSM is detected
HKLM\SYSTEM\CurrentControlSet\Services \mpio\Parameters\RetryInterval	1	When MPIO support is specified, except if Data ONTAP DSM is detected
HKLM\SYSTEM\CurrentControlSet\Services \msdsm\Parameters\RetryInterval	1	When MPIO support is specified and your server is Windows Server 2016, 2012 R2, 2012, 2008 R2, or 2008, except if Data ONTAP DSM is detected
HKLM\SYSTEM\CurrentControlSet\Services \vnetapp\Parameters\RetryInterval	1	When MPIO support is specified, except if Data ONTAP DSM is detected
HKLM\SYSTEM\CurrentControlSet \Services\disk\TimeOutValue	120	When no MPIO support is selected
	60	When MPIO support is specified

Registry key	Value	Set registry key…
HKLM\SYSTEM\CurrentControlSet\Services\mpio\Parameters\UseCustomPathRecoveryInterval	1	When your server is Windows Server 2016, 2012 R2, 2012, 2008 R2, or 2008

See the Microsoft documents for the registry parameter details.

Learn about the FC HBA values set by Windows Host Utilities

The Windows Host Utilities installer sets the required timeout values for Emulex and QLogic FC HBAs on systems using FC.

The installer sets the following parameters for Emulex FC HBAs:

When you select MPIO

Property type	Property value
LinkTimeOut	1
NodeTimeOut	10

When you don't select MPIO

Property type	Property value
LinkTimeOut	30
NodeTimeOut	120

The installer sets the following parameters for QLogic FC HBAs:

When you select MPIO

Property type	Property value
LinkDownTimeOut	1
PortDownRetryCount	10

When you don't select MPIO

Property type	Property value
LinkDownTimeOut	30
PortDownRetryCount	120

The names of the parameters might vary slightly depending on the program. For example, in the QLogic QConvergeConsole program, the parameter is displayed as `Link Down Timeout`. The Host Utilities `fcconfig.ini` file displays this parameter as either `LinkDownTimeOut` or `MpioLinkDownTimeOut`, depending on whether MPIO is specified. However, all of these names refer to the same HBA parameter. See Emulex or QLogic to learn more about the timeout parameters.

Learn about the Host Utilities changes to FC HBA driver settings

During installation of the required Emulex or QLogic HBA drivers on an FC system, Windows Host Utilities checks several parameters and, in some cases, modifies them.

The Windows Host Utilities sets values for the following parameters if MS DSM for Windows MPIO is detected:

- **LinkTimeOut**: Defines the length of time in seconds that the host port waits before resuming I/O after a physical link is down.
- **NodeTimeOut**: Defines the length of time in seconds before the host port recognizes that a connection to the target device is down.

When troubleshooting HBA issues, check that these settings have the correct values. The correct values depend on two factors:

- The HBA vendor
- Whether you are using MPIO software.

You can correct the HBA settings by running the Repair option in the Windows Host Utilities installer.

Emulex HBA drivers

Verify the Emulex HBA driver settings on FC systems. These settings must exist for each port on the HBA.

Steps

1. Open OnCommand Manager.
2. Select the appropriate HBA from the list and select the **Driver Parameters** tab.

   The driver parameters appear.

   a. If you are using MPIO software, ensure that you have the following driver settings:
      - LinkTimeOut - 1
      - NodeTimeOut - 10
   b. If you aren't using MPIO software, ensure that you have the following driver settings:
      - LinkTimeOut - 30
      - NodeTimeOut - 120

QLogic HBA drivers

Verify the QLogic HBA driver settings on FC systems. These settings must exist for each port on the HBA.

Steps

1. Open QConvergeConsole, and then select **Connect** on the toolbar.

   The **Connect to Host** dialog box appears.

2. Select the appropriate host from the list, and then select **Connect**.

   A list of HBAs appears in the FC HBA pane.

3. Select the appropriate HBA port from the list, and then select the **Settings** tab.
4. Select **Advanced HBA Port Settings** from the **Select Settings** section.
5. If you are using MPIO software, verify that you have the following driver settings:
   - Link Down Timeout (linkdwnto) - 1
   - Port Down Retry Count (portdwnrc) - 10
6. If you aren't using MPIO software, verify that you have the following driver settings:
   - Link Down Timeout (linkdwnto) - 30
   - Port Down Retry Count (portdwnrc) - 120

Upgrade Windows Host Utilities for ONTAP storage

You can upgrade Windows Host Utilities software in your ONTAP storage configuration by following the installation instructions interactively or by using the Windows command line. You need to install the new Windows Host Utilities installation software on a path that is accessible to the Windows host.

Upgrade interactively

Upgrade the Windows Host Utilities software interactively by running the Host Utilities installation program and following the prompts.

Steps

1. Change to the directory where you downloaded the executable file.
2. Run the executable file and follow the instructions on the screen.
3. Reboot the Windows host when prompted.
4. After the reboot completes, check the host utility version:
   a. Open **Control Panel**.
   b. Go to **Program and features** and check the host utility version.

Upgrade noninteractively

Perform a noninteractive Windows Host Utilities software upgrade by using the Windows command line.

Steps

1. Enter the following command at the Windows command prompt:

```
msiexec /i installer.msi /quiet MULTIPATHING= {0 | 1}
[INSTALLDIR=inst_path]
```

   - `installer` is the name of the `.msi` file for your CPU architecture.
   - MULTIPATHING specifies whether Microsoft Multipath I/O (MPIO) support is installed. The allowed values are "0" for no and "1" for yes.
   - `inst_path` is the path where the Host Utilities files are installed. The default path is `C:\Program Files\NetApp\Windows Host Utilities\`.

> To see the standard Microsoft Installer (MSI) options for logging and other functions, enter `msiexec /help` at the Windows command prompt. For example, the `msiexec /i install.msi /quiet /l*v <install.log> LOGVERBOSE=1` command displays logging information.

The system automatically reboots when the installation is complete.

Repair and remove the Windows Host Utilities for ONTAP storage

Use the **Repair** option in the Host Utilities installation program to update the Host bus adapter (HBA) and Windows registry settings. You can also remove the Host Utilities entirely, either interactively or from the Windows command line.

Repair or remove interactively

The **Repair** option updates the Windows registry and FC HBAs with the required settings. You can also remove the Host Utilities entirely.

Steps

1. Open Windows **Programs and Features** (Windows Server 2012 R2, Windows Server 2016, Windows Server 2019, and Windows 2022).
2. Select **NetApp Windows Host Utilities**.
3. Select **Change**.
4. Select **Repair** or **Remove**, as needed.
5. Follow the instructions on the screen.

Repair or remove noninteractively

The **Repair** option updates the Windows registry and FC HBAs with the required settings. You can also remove the Host Utilities entirely from a Windows command line.

Steps

1. Repair Windows Host Utilities:

```
msiexec /f installer.msi [/quiet]
```

   - `/f` repairs the installation.
   - `installer.msi` is the name of the Windows Host Utilities installation program on your system.
   - `/quiet` suppresses all feedback and reboots the system automatically without prompting when the command completes.

Troubleshoot issues for Windows Host Utilities with ONTAP storage configuration

Use the general troubleshooting techniques to investigate issues that might occur for a Windows Host Utilities with ONTAP storage configuration. You should also check the Windows Host Utilities Release Notes for known issues and solutions.

The following is a list of the different areas you can investigate for potential interoperability issues:

- To identify potential interoperability issues, confirm that the Host Utilities support your combination of host operating system software, host hardware, ONTAP software, and storage system hardware. See the Interoperability Matrix Tool for more information.
- Verify that you have the correct iSCSI configuration.
- If iSCSI LUNs are not available after a reboot, verify that the target is listed as persistent on the **Persistent Targets** tab of the Microsoft iSCSI initiator GUI.
- If applications using the LUNs display errors on startup, verify that the applications are configured to depend on the iSCSI service.
- For FC paths to storage controllers running ONTAP, verify that the FC switches are zoned using the WWPNs of the target LIFs, not the WWPNs of the physical ports on the node.

- Review the Release Notes for Windows Host Utilities to check for known issues. The Release Notes include a list of known issues and limitations.
- Review the troubleshooting information in the ONTAP SAN administration documentation.
- Search NetApp Bugs Online for recently discovered issues.
  - In the Bug Type field under Advanced Search, select **iSCSI - Windows** and then select **Go**. You should repeat the search for Bug Type **FCP -Windows**.
- Collect information about your system.
- Record any error messages that are displayed on the host or storage system console.
- Collect the host and storage system log files.
- Record the symptoms of the problem and any changes made to the host or storage system just before the problem appeared.
- If you are unable to resolve the problem, contact NetApp technical support for assistance.

Configure hosts with FCP and iSCSI

Overview

You can configure certain SAN hosts for FCP or iSCSI with ONTAP as the target. First you install the relevant operating system host utility package, which includes the SAN tool kit. You then verify the multipath settings for the NetApp ONTAP LUNs.

AIX and PowerVM/VIOS

Configure AIX 7.3/VIOS 4.x for FCP and iSCSI with ONTAP storage

The AIX Host Utilities software provides management and diagnostic tools for AIX hosts that are connected to ONTAP storage. When you install the AIX Host Utilities on an AIX 7.3/VIOS 4.x host, you can use the Host Utilities to help you manage FCP and iSCSI protocol operations with ONTAP LUNs.

Step 1: Optionally, enable SAN booting

You can configure your host to use SAN booting to simplify deployment and improve scalability. If your configuration doesn't support SAN booting, you can use a local boot.

SAN boot

SAN booting is the process of setting up a SAN-attached disk (a LUN) as a boot device for an AIX/PowerVM host host. You can set up a SAN boot LUN to work in a AIX Multipath I/O (MPIO) environment that is using the FC protocol and running AIX Host Utilities with either the FC or FCoE protocol. The method you use to create a SAN boot LUN and install a new OS image in an AIX MPIO environment depends on the protocol that you are using.

Steps

1. Use the Interoperability Matrix Tool to verify that your AIX OS, protocol, and ONTAP version support SAN booting.
2. Follow the best practices for setting up a SAN boot in the vendor documentation.

Local boot

Perform a local boot by installing the AIX OS on the local hard disk, for example, on an SSD, SATA, or RAID.

Step 2: Install the AIX Host Utilities

NetApp strongly recommends installing the AIX Host Utilities to support ONTAP LUN management and assist technical support with gathering configuration data. The MPIO package from the Host Utilities provides MPIO support for AIX and VIOS.

Installing the AIX Host Utilities provides additional timeout settings on your AIX host.

Install AIX Host Utilities 8.0.

Step 3: Confirm the multipath configuration for your host

You can use multipathing with AIX and PowerVM to manage ONTAP LUNs.

Multipathing allows you to configure multiple network paths between the host and storage system. If one path fails, traffic continues with the remaining paths. The AIX and PowerVM environments of the Host Utilities use the AIXs native multipathing solution (MPIO).

The Path Control Module (PCM) is responsible for controlling multiple paths for an AIX host. The PCM is a storage vendor supplied code that handles path management and is installed and enabled during the Host Utilities installation.

To ensure that multipathing is configured correctly for your host, verify that you have the NetApp recommended settings configured for your ONTAP LUNs.

Steps

1. Verify that "MPIO NetApp" is available. "MPIO NetApp" is loaded during the AIX Host Utilities installation and becomes available after you reboot the host.

   ```
   lsdev -Cc disk
   ```

   Example output

   ```
   hdisk1 Available 00-00-02 MPIO NetApp FCP Default PCM Disk
   ```

2. The AIX Host Utilities load the following parameter settings for ONTAP LUNs.

Show parameter settings

Parameter	Environment	Value for AIX	Note
algorithm	MPIO	round_robin	Set by Host Utilities
hcheck_cmd	MPIO	inquiry	Set by Host Utilities
hcheck_interval	MPIO	30	Set by Host Utilities
hcheck_mode	MPIO	nonactive	Set by Host Utilities
lun_reset_spt	MPIO / non-MPIO	yes	Set by Host Utilities
max_transfer	MPIO / non-MPIO	FC LUNs: 0x100000 bytes	Set by Host Utilities
qfull_dly	MPIO / non-MPIO	2-second delay	Set by Host Utilities
queue_depth	MPIO / non-MPIO	64	Set by Host Utilities
reserve_policy	MPIO / non-MPIO	no_reserve	Set by Host Utilities
re_timeout (disk)	MPIO / non-MPIO	30 seconds	Uses OS Default values
dyntrk	MPIO / non-MPIO	Yes	Uses OS Default values
fc_err_recov	MPIO / non-MPIO	Fast_fail	Uses OS Default values
q_type	MPIO / non-MPIO	simple	Uses OS Default values
num_cmd_elems	MPIO / non-MPIO	1024 for AIX 3072 for VIOS	FC EN1B, FC EN1C
num_cmd_elems	MPIO / non-MPIO	1024 for AIX	FC EN0G

3. Configure the following settings to optimize I/O operations for FC.

Parameter	AIX versions	AIX OS default value	NetApp recommended value
rw_timeout (disk)	AIX 7.3TL3	NPIV:30 seconds, vSCSI:45 seconds	NPIV:30 seconds, vSCSI:120 seconds
	AIX 7.2TL5	NPIV:30 seconds, vSCSI:45 seconds	NPIV:30 seconds, vSCSI:120 seconds
	VIOS 3.1	30 seconds	30 seconds
	VIOS 4.1	30 seconds	30 seconds

4. Configure the following settings to optimize I/O operations for iSCSI.

Parameter	AIX versions	AIX OS default value	NetApp recommended value
rw_timeout (disk)	AIX 7.3TL3	vSCSI:45 seconds	vSCSI:120 seconds
	AIX 7.2TL5	vSCSI:45 seconds	vSCSI:120 seconds
	VIOS 3.1	120 seconds	30 seconds
	VIOS 4.1	120 seconds	30 seconds
	All AIX7.2 and AIX 7.3 standalone	120 seconds	30 seconds
isw_err_recov (iscsi0)	All AIX7.2 and AIX 7.3 standalone	delayed_fail	fast_fail

5. If your storage configuration includes MetroCluster or SnapMirror active sync, change the default settings:

MetroCluster

By default, the AIX OS enforces a shorter I/O timeout when there are no available paths to a LUN. This might occur in configurations that include single-switch SAN fabric and in MetroCluster configurations that experience unplanned failovers. For additional information and recommended changes to default settings, see the Knowledge Base article What are AIX Host support considerations in a MetroCluster configuration?.

SnapMirror active sync

Beginning with ONTAP 9.11.1, SnapMirror active sync is supported for an AIX host. The primary cluster in an AIX configuration is the "active" cluster.

In an AIX configuration, failovers are disruptive. With each failover, you need to perform a re-scan on the host for I/O operations to resume.

Refer to the Knowledge Base article How to configure an AIX host for SnapMirror active sync.

6. Verify the parameter settings and that multiple paths are listed for an ONTAP LUN:

```
lsmpio
```

In the following example for an AFF or FAS system, the PCM is listed for NetApp.

Show example

```
# lsmpio -l hdisk1
name    path_id  status   path_status  parent  connection

hdisk1  0        Enabled  Non          fscsi6
203200a098ba7afe,5b000000000000
hdisk1  1        Enabled  Non          fscsi8
203100a098ba7afe,5b000000000000
hdisk1  2        Enabled  Sel,Opt      fscsi6
203000a098ba7afe,5b000000000000
hdisk1  3        Enabled  Sel,Opt      fscsi8
203800a098ba7afe,5b000000000000
#
lsattr -El hdisk1
PCM             PCM/friend/NetAppDefaultPCM Path Control Module
False
PR_key_value    0x6d0000000002              Persistant Reserve Key
Value            True
algorithm       round_robin                 Algorithm
True
clr_q           no                          Device CLEARS its Queue
on error        True
dist_err_pcnt   0                           Distributed Error Sample
Time           True
dist_tw_width   50                          Distributed Error Sample
Time           True
hcheck_cmd      inquiry                     Health Check Command
True
hcheck_interval 30                          Health Check Interval
True
hcheck_mode     nonactive                   Health Check Mode
True
location                                    Location Label
True
lun_id          0x5b000000000000            Logical Unit Number ID
False
lun_reset_spt   yes                         LUN Level Reset
True
max_transfer    0x100000                    Maximum TRANSFER Size
True
node_name       0x204800a098ba7afe          FC Node Name
False
pvid            none                        Physical volume
identifier              False
q_err           yes                         Use QERR bit
```

```
True
q_type          simple                          Queuing TYPE
True
qfull_dly       2                               Delay in seconds for
SCSI TASK SET FULL True
queue_depth     64                              Queue DEPTH
True
reassign_to     120                             REASSIGN time out value
True
reserve_policy  PR_shared                       Reserve Policy
True
rw_timeout      30                              READ/WRITE time out
value               True
scsi_id         0xec409                         SCSI ID
False
start_timeout   60                              START unit time out
value               True
timeout_policy  fail_path                       Active/Passive Disk Path
Control Module True
ww_name         0x203200a098ba7afe              FC World Wide Name
False
```

7. Verify the path status for ONTAP LUNs:

```
sanlun lun show
```

The following example outputs show the correct path status for ONTAP LUNs in an ASA, AFF, or FAS configuration.

ASA configurations

An ASA configuration optimizes all paths to a given LUN, keeping them active ("primary"). This improves performance by serving I/O operations through all paths at the same time.

Show example

```
# sanlun lun show -p |grep -p hdisk78
                    ONTAP Path:
vs_aix_clus:/vol/chataix_205p2_vol_en_1_7/jfs_205p2_lun_en
                           LUN: 37
                      LUN Size: 15g
                   Host Device: hdisk78
                          Mode: C
            Multipath Provider: AIX Native
        Multipathing Algorithm: round_robin
------ ------- ------ ------- --------- ----------
host   vserver  AIX                      AIX MPIO
path   path     MPIO   host    vserver     path
state  type     path   adapter LIF       priority
------ ------- ------ ------- --------- ----------
up     primary  path0  fcs0    fc_aix_1     1
up     primary  path1  fcs0    fc_aix_2     1
up     primary  path2  fcs1    fc_aix_3     1
up     primary  path3  fcs1    fc_aix_4     1
```

AFF or FAS configuration

An AFF or FAS configuration should have two groups of paths with higher and lower priorities. Higher priority Active/Optimized paths are served by the controller where the aggregate is located. Lower priority paths are active but non-optimized because they are served by a different controller. Non-optimized paths are only used when optimized paths aren’t available.

The following example displays the correct output for an ONTAP LUN with two Active/Optimized ("primary") paths and two Active/Non-Optimized ("secondary") paths:

Show example

```
# sanlun lun show -p |grep -p hdisk78
                    ONTAP Path:
vs_aix_clus:/vol/chataix_205p2_vol_en_1_7/jfs_205p2_lun_en
                           LUN: 37
                      LUN Size: 15g
                   Host Device: hdisk78
                          Mode: C
              Multipath Provider: AIX Native
        Multipathing Algorithm: round_robin
------- ---------- ------ ------- ---------- ----------
host    vserver    AIX                        AIX MPIO
path    path       MPIO   host    vserver         path
state   type       path   adapter LIF         priority
------- ---------- ------ ------- ---------- ----------
up      secondary  path0  fcs0    fc_aix_1         1
up      primary    path1  fcs0    fc_aix_2         1
up      primary    path2  fcs1    fc_aix_3         1
up      secondary  path3  fcs1    fc_aix_4         1
```

Step 4: Review the known issues

There are no known issues.

What's next?

Learn about using the AIX Host Utilities tool.

Configure AIX 7.2/VIOS 3.1 for FCP and iSCSI with ONTAP storage

The AIX Host Utilities software provides management and diagnostic tools for AIX hosts that are connected to ONTAP storage. When you install the AIX Host Utilities on an AIX 7.2 and/or PowerVM (VIOS 3.1) host, you can use the Host Utilities to help you manage FCP and iSCSI protocol operations with ONTAP LUNs.

Step 1: Optionally, enable SAN booting

You can configure your host to use SAN booting to simplify deployment and improve scalability. If your configuration doesn't support SAN booting, you can use a local boot.

SAN boot

SAN booting is the process of setting up a SAN-attached disk (a LUN) as a boot device for an AIX/PowerVM host host. You can set up a SAN boot LUN to work in a AIX Multipath I/O (MPIO) environment that is using the FC protocol and running AIX Host Utilities with either the FC or FCoE protocol. The method you use to create a SAN boot LUN and install a new OS image in an AIX MPIO environment depends on the protocol that you are using.

Steps

1. Use the Interoperability Matrix Tool to verify that your AIX OS, protocol, and ONTAP version support SAN booting.
2. Follow the best practices for setting up a SAN boot in the vendor documentation.

Local boot

Perform a local boot by installing the AIX OS on the local hard disk, for example, on an SSD, SATA, or RAID.

Step 2: Install the AIX Host Utilities

NetApp strongly recommends installing the AIX Host Utilities to support ONTAP LUN management and assist technical support with gathering configuration data. The MPIO package from the Host Utilities provides MPIO support for AIX and VIOS.

Installing the AIX Host Utilities provides additional timeout settings on your AIX host.

Install AIX Host Utilities 6.1.

Step 3: Confirm the multipath configuration for your host

You can use multipathing with AIX 7.2 and PowerVM to manage ONTAP LUNs.

Multipathing allows you to configure multiple network paths between the host and storage system. If one path fails, traffic continues with the remaining paths. The AIX and PowerVM environments of the Host Utilities use the AIXs native multipathing solution (MPIO).

The Path Control Module (PCM) is responsible for controlling multiple paths for an AIX host. The PCM is a storage vendor supplied code that handles path management and is installed and enabled during the Host Utilities installation.

To ensure that multipathing is configured correctly for your host, verify that you have the NetApp recommended settings configured for your ONTAP LUNs.

1. The AIX Host Utilities load the following parameter settings for ONTAP LUNs.

Show parameter settings

Parameter	Environment	Value for AIX	Note
algorithm	MPIO	round_robin	Set by Host Utilities
hcheck_cmd	MPIO	inquiry	Set by Host Utilities
hcheck_interval	MPIO	30	Set by Host Utilities
hcheck_mode	MPIO	nonactive	Set by Host Utilities
lun_reset_spt	MPIO / non-MPIO	yes	Set by Host Utilities
max_transfer	MPIO / non-MPIO	FC LUNs: 0x100000 bytes	Set by Host Utilities
qfull_dly	MPIO / non-MPIO	2-second delay	Set by Host Utilities
queue_depth	MPIO / non-MPIO	64	Set by Host Utilities
reserve_policy	MPIO / non-MPIO	no_reserve	Set by Host Utilities
re_timeout (disk)	MPIO / non-MPIO	30 seconds	Uses OS Default values
dyntrk	MPIO / non-MPIO	Yes	Uses OS Default values
fc_err_recov	MPIO / non-MPIO	Fast_fail	Uses OS Default values
q_type	MPIO / non-MPIO	simple	Uses OS Default values
num_cmd_elems	MPIO / non-MPIO	1024 for AIX 3072 for VIOS	FC EN1B, FC EN1C
num_cmd_elems	MPIO / non-MPIO	1024 for AIX	FC EN0G

2. If your storage configuration includes MetroCluster or SnapMirror active sync, change the default settings:

MetroCluster

By default, the AIX OS enforces a shorter I/O timeout when there are no available paths to a LUN. This might occur in configurations that include single-switch SAN fabric and in MetroCluster configurations that experience unplanned failovers. For additional information and recommended changes to default settings, see the Knowledge Base article What are AIX Host support considerations in a MetroCluster configuration?.

SnapMirror active sync

Beginning with ONTAP 9.11.1, SnapMirror active sync is supported for an AIX host. The primary cluster in an AIX configuration is the "active" cluster.

In an AIX configuration, failovers are disruptive. With each failover, you need to perform a re-scan on the host for I/O operations to resume.

Refer to the Knowledge Base article How to configure an AIX host for SnapMirror active sync.

3. Verify the path status for ONTAP LUNs:

```
sanlun lun show
```

The following example outputs show the correct path status for ONTAP LUNs in an ASA, AFF, or FAS configuration.

ASA configurations

An ASA configuration optimizes all paths to a given LUN, keeping them active ("primary"). This improves performance by serving I/O operations through all paths at the same time.

Show example

```
# sanlun lun show -p |grep -p hdisk78
                    ONTAP Path:
vs_aix_clus:/vol/chataix_205p2_vol_en_1_7/jfs_205p2_lun_en
                           LUN: 37
                      LUN Size: 15g
                   Host Device: hdisk78
                          Mode: C
            Multipath Provider: AIX Native
        Multipathing Algorithm: round_robin
------ ------- ------ ------- --------- ----------
host   vserver  AIX                      AIX MPIO
path   path     MPIO   host    vserver     path
state  type     path   adapter LIF       priority
------ ------- ------ ------- --------- ----------
up     primary  path0  fcs0    fc_aix_1     1
up     primary  path1  fcs0    fc_aix_2     1
up     primary  path2  fcs1    fc_aix_3     1
up     primary  path3  fcs1    fc_aix_4     1
```

AFF or FAS configuration

An AFF or FAS configuration should have two groups of paths with higher and lower priorities. Higher priority Active/Optimized paths are served by the controller where the aggregate is located. Lower priority paths are active but non-optimized because they are served by a different controller. Non-optimized paths are only used when optimized paths aren’t available.

The following example displays the correct output for an ONTAP LUN with two Active/Optimized ("primary") paths and two Active/Non-Optimized ("secondary") paths:

Show example

```
# sanlun lun show -p |grep -p hdisk78
                    ONTAP Path:
vs_aix_clus:/vol/chataix_205p2_vol_en_1_7/jfs_205p2_lun_en
                           LUN: 37
                      LUN Size: 15g
                   Host Device: hdisk78
                          Mode: C
              Multipath Provider: AIX Native
        Multipathing Algorithm: round_robin
------- ---------- ------ ------- ---------- ----------
host    vserver    AIX                        AIX MPIO
path    path       MPIO   host    vserver         path
state   type       path   adapter LIF         priority
------- ---------- ------ ------- ---------- ----------
up      secondary  path0  fcs0    fc_aix_1        1
up      primary    path1  fcs0    fc_aix_2        1
up      primary    path2  fcs1    fc_aix_3        1
up      secondary  path3  fcs1    fc_aix_4        1
```

Step 4: Review the known issues

Known issues

The IBM AIX 7.2 and/or PowerVM (VIOS 3.1) with ONTAP storage release has the following known issues:

NetApp Bug ID	Title	Description	Partner ID
1416221	AIX 7200-05-01 encountered I/O disruption on virtual iSCSI disks(VIOS 3.1.1.x) during storage failover	I/O disruption can happen during storage failover operations on AIX 7.2 TL5 hosts on the virtual iSCSI disks mapped through the VIOS 3.1.1.x. By default, the `rw_timeout` value of the virtual iSCSI disks (hdisk) on VIOC will be 45 seconds. If an I/O delay greater than 45 seconds happens during storage failover, an I/O failure might occur. To avoid this situation, refer to the workaround mentioned in the BURT. As per IBM, after applying APAR - IJ34739 (upcoming release) we can dynamically change the rw_timeout value using the `chdev` command.	NA
1414700	AIX 7.2 TL04 encountered I/O disruption on virtual iSCSI disks(VIOS 3.1.1.x) during storage failover	I/O disruption can happen during storage failover operations on AIX 7.2 TL4 hosts on the virtual iSCSI disks mapped through the VIOS 3.1.1.x. By default, the `rw_timeout` value of vSCSI adapter on VIOC is 45 seconds. If an I/O delay of more than 45 seconds happens during a storage failover, I/O failure might occur. To avoid this situation, refer to the workaround mentioned in the BURT.	NA
1307653	I/O issues occur on VIOS 3.1.1.10 during SFO faults and straight I/O	On VIOS 3.1.1, I/O failures might occur on NPIV client disks that are backed by 16 GB or 32 GB FC adapters. Additionally, the `vfchost` driver might stop processing I/O requests from the client. Applying IBM APAR IJ22290 IBM APAR IJ23222 fixes the issue.	NA

What's next?

Learn about using the AIX Host Utilities tool.

Configure AIX 7.1 for FCP and iSCSI with ONTAP storage

The AIX Host Utilities software provides management and diagnostic tools for AIX hosts that are connected to ONTAP storage. When you install the AIX Host Utilities on an AIX 7.1 host, you can use the Host Utilities to help you manage FCP and iSCSI protocol operations with ONTAP LUNs.

Step 1: Optionally, enable SAN booting

You can configure your host to use SAN booting to simplify deployment and improve scalability. If your configuration doesn't support SAN booting, you can use a local boot.

SAN boot

SAN booting is the process of setting up a SAN-attached disk (a LUN) as a boot device for an AIX/PowerVM host host. You can set up a SAN boot LUN to work in a AIX Multipath I/O (MPIO) environment that is using the FC protocol and running AIX Host Utilities with either the FC or FCoE protocol. The method you use to create a SAN boot LUN and install a new OS image in an AIX MPIO environment depends on the protocol that you are using.

Steps

1. Use the Interoperability Matrix Tool to verify that your AIX OS, protocol, and ONTAP version support SAN booting.
2. Follow the best practices for setting up a SAN boot in the vendor documentation.

Local boot

Perform a local boot by installing the AIX OS on the local hard disk, for example, on an SSD, SATA, or RAID.

Step 2: Install the AIX Host Utilities

NetApp strongly recommends installing the AIX Host Utilities to support ONTAP LUN management and assist technical support with gathering configuration data. The MPIO package from the Host Utilities provides MPIO support for AIX and VIOS.

Installing the AIX Host Utilities provides additional timeout settings on your AIX host.

Install AIX Host Utilities 6.1.

Step 3: Confirm the multipath configuration for your host

You can use multipathing with an AIX 7.1 host to manage ONTAP LUNs.

Multipathing allows you to configure multiple network paths between the host and storage system. If one path fails, traffic continues with the remaining paths. The AIX and PowerVM environments of the Host Utilities use the AIXs native multipathing solution (MPIO).

The Path Control Module (PCM) is responsible for controlling multiple paths for an AIX host. The PCM is a storage vendor supplied code that handles path management and is installed and enabled during the Host Utilities installation.

To ensure that multipathing is configured correctly for your host, verify that you have the NetApp recommended settings configured for your ONTAP LUNs.

Steps

1. The AIX Host Utilities load the following parameter settings for ONTAP LUNs.

   Show parameter settings

Parameter	Environment	Value for AIX	Note
algorithm	MPIO	round_robin	Set by Host Utilities
hcheck_cmd	MPIO	inquiry	Set by Host Utilities
hcheck_interval	MPIO	30	Set by Host Utilities
hcheck_mode	MPIO	nonactive	Set by Host Utilities
lun_reset_spt	MPIO / non-MPIO	yes	Set by Host Utilities
max_transfer	MPIO / non-MPIO	FC LUNs: 0x100000 bytes	Set by Host Utilities
qfull_dly	MPIO / non-MPIO	2-second delay	Set by Host Utilities
queue_depth	MPIO / non-MPIO	64	Set by Host Utilities
reserve_policy	MPIO / non-MPIO	no_reserve	Set by Host Utilities
re_timeout (disk)	MPIO / non-MPIO	30 seconds	Uses OS Default values
dyntrk	MPIO / non-MPIO	Yes	Uses OS Default values
fc_err_recov	MPIO / non-MPIO	Fast_fail	Uses OS Default values
q_type	MPIO / non-MPIO	simple	Uses OS Default values
num_cmd_elems	MPIO / non-MPIO	1024 for AIX	FC EN1B, FC EN1C
num_cmd_elems	MPIO / non-MPIO	500 for AIX (standalone/physical) 200 for VIOC	FC EN0G

2. If your storage configuration includes MetroCluster or SnapMirror active sync, change the default settings:

MetroCluster

By default, the AIX OS enforces a shorter I/O timeout when there are no available paths to a LUN. This might occur in configurations that include single-switch SAN fabric and in MetroCluster configurations that experience unplanned failovers. For additional information and recommended changes to default settings, see the Knowledge Base article What are AIX Host support considerations in a MetroCluster configuration?.

SnapMirror active sync

Beginning with ONTAP 9.11.1, SnapMirror active sync is supported for an AIX host. The primary cluster in an AIX configuration is the "active" cluster.

In an AIX configuration, failovers are disruptive. With each failover, you need to perform a re-scan on the host for I/O operations to resume.

Refer to the Knowledge Base article How to configure an AIX host for SnapMirror active sync.

3. Verify the path status for ONTAP LUNs:

```
sanlun lun show
```

The following example outputs show the correct path status for ONTAP LUNs in an ASA, AFF, or FAS configuration.

ASA configurations

An ASA configuration optimizes all paths to a given LUN, keeping them active ("primary"). This improves performance by serving I/O operations through all paths at the same time.

Show example

```
# sanlun lun show -p |grep -p hdisk78
                    ONTAP Path:
vs_aix_clus:/vol/chataix_205p2_vol_en_1_7/jfs_205p2_lun_en
                           LUN: 37
                      LUN Size: 15g
                   Host Device: hdisk78
                          Mode: C
            Multipath Provider: AIX Native
        Multipathing Algorithm: round_robin
------ ------- ------ ------- --------- ----------
host   vserver  AIX                      AIX MPIO
path   path     MPIO   host    vserver     path
state  type     path   adapter LIF       priority
------ ------- ------ ------- --------- ----------
up     primary  path0  fcs0    fc_aix_1     1
up     primary  path1  fcs0    fc_aix_2     1
up     primary  path2  fcs1    fc_aix_3     1
up     primary  path3  fcs1    fc_aix_4     1
```

AFF or FAS configuration

An AFF or FAS configuration should have two groups of paths with higher and lower priorities. Higher priority Active/Optimized paths are served by the controller where the aggregate is located. Lower priority paths are active but non-optimized because they are served by a different controller. Non-optimized paths are only used when optimized paths aren’t available.

The following example displays the correct output for an ONTAP LUN with two Active/Optimized ("primary") paths and two Active/Non-Optimized ("secondary") paths:

Show example

```
# sanlun lun show -p |grep -p hdisk78
                    ONTAP Path:
vs_aix_clus:/vol/chataix_205p2_vol_en_1_7/jfs_205p2_lun_en
                           LUN: 37
                      LUN Size: 15g
                   Host Device: hdisk78
                          Mode: C
            Multipath Provider: AIX Native
        Multipathing Algorithm: round_robin
------- ---------- ------ ------- ---------- ----------
host    vserver    AIX                        AIX MPIO
path    path       MPIO   host    vserver         path
state   type       path   adapter LIF         priority
------- ---------- ------ ------- ---------- ----------
up      secondary  path0  fcs0    fc_aix_1         1
up      primary    path1  fcs0    fc_aix_2         1
up      primary    path2  fcs1    fc_aix_3         1
up      secondary  path3  fcs1    fc_aix_4         1
```

Step 4: Review the known issues

The AIX 7.1 with ONTAP storage release has no known issues.

What's next?

Learn about using the AIX Host Utilities tool.

CentOS

Configure CentOS 8.x for FCP and iSCSI for ONTAP storage

The Linux Host Utilities software provides management and diagnostic tools for Linux hosts that are connected to ONTAP storage. When you install the Linux Host Utilities on a CentOS 8.x host, you can use the Host Utilities to help you manage FCP and iSCSI protocol operations with ONTAP LUNs.

Step 1: Optionally, enable SAN booting

You can configure your host to use SAN booting to simplify deployment and improve scalability.

Before you begin

Use the Interoperability Matrix Tool to verify that your Linux OS, host bus adapter (HBA), HBA firmware, HBA boot BIOS, and ONTAP version support SAN booting.

Steps

1. Create a SAN boot LUN and map it to the host.
2. Enable SAN booting in the server BIOS for the ports to which the SAN boot LUN is mapped.

   For information on how to enable the HBA BIOS, see your vendor-specific documentation.

3. Verify that the configuration was successful by rebooting the host and verifying that the OS is up and running.

Step 2: Install the Linux Host Utilities

NetApp strongly recommends installing the Linux Host Utilities to support ONTAP LUN management and assist technical support with gathering configuration data.

Install Linux Host Utilities 8.0.

Installing the Linux Host Utilities doesn't change any host timeout settings on your Linux host.

Step 3: Confirm the multipath configuration for your host

You can use multipathing with CentOS 8.x to manage ONTAP LUNs.

To ensure that multipathing is configured correctly for your host, verify that the `/etc/multipath.conf` file is defined and that you have the NetApp recommended settings configured for your ONTAP LUNs.

Steps

1. Verify that the `/etc/multipath.conf` file exits. If the file doesn't exist, create an empty, zero-byte file:

   ```
   touch /etc/multipath.conf
   ```

2. The first time the `multipath.conf` file is created, you might need to enable and start the multipath services to load the recommended settings:

   ```
   chkconfig multipathd on
   ```

   ```
   /etc/init.d/multipathd start
   ```

3. Each time you boot the host, the empty `/etc/multipath.conf` zero-byte file automatically loads the NetApp recommended host multipath parameters as the default settings. You shouldn't need to make changes to the `/etc/multipath.conf` file for your host because the operating system is compiled with the multipath parameters that recognize and manage ONTAP LUNs correctly.

   The following table shows the Linux OS native compiled multipath parameter settings for ONTAP LUNs.

Show parameter settings

Parameter	Setting
detect_prio	yes
dev_loss_tmo	"infinity"
failback	immediate
fast_io_fail_tmo	5
features	"2 pg_init_retries 50"
flush_on_last_del	"yes"
hardware_handler	"0"
no_path_retry	queue
path_checker	"tur"
path_grouping_policy	"group_by_prio"
path_selector	"service-time 0"
polling_interval	5
prio	"ontap"
product	LUN
retain_attached_hw_handler	yes
rr_weight	"uniform"
user_friendly_names	no
vendor	NETAPP

4. Verify the parameter settings and path status for your ONTAP LUNs:

```
multipath -ll
```

The default multipath parameters support ASA, AFF, and FAS configurations. In these configurations, a single ONTAP LUN shouldn’t require more than four paths. Having more than four paths can cause problems during a storage failure.

The following example outputs show the correct parameter settings and path status for ONTAP LUNs in an ASA, AFF, or FAS configuration.

ASA configuration

An ASA configuration optimizes all paths to a given LUN, keeping them active. This improves performance by serving I/O operations through all paths at the same time.

Show example

```
multipath -ll
3600a098038303634722b4d59646c4436 dm-28 NETAPP,LUN C-Mode
size=80G features='3 queue_if_no_path pg_init_retries 50'
hwhandler='1 alua' wp=rw
`-+- policy='service-time 0' prio=50 status=active
  |- 11:0:7:1    sdfi   130:64   active ready running
  |- 11:0:9:1    sdiy  8:288     active ready running
  |- 11:0:10:1  sdml 69:464   active ready running
  |- 11:0:11:1  sdpt  131:304  active ready running
```

AFF or FAS configuration

An AFF or FAS configuration should have two groups of paths with higher and lower priorities. Higher priority Active/Optimized paths are served by the controller where the aggregate is located. Lower priority paths are active but non-optimized because they are served by a different controller. Non-optimized paths are only used when optimized paths aren’t available.

The following example displays the output for an ONTAP LUN with two Active/Optimized paths and two Active/Non-Optimized paths:

Show example

```
multipath -ll
3600a098038303634722b4d59646c4436 dm-28 NETAPP,LUN C-Mode
size=80G features='3 queue_if_no_path pg_init_retries 50'
hwhandler='1 alua' wp=rw
|-+- policy='service-time 0' prio=50 status=active
| |- 16:0:6:35 sdwb  69:624  active ready running
| |- 16:0:5:35 sdun  66:752  active ready running
`-+- policy='service-time 0' prio=10 status=enabled
  |- 15:0:0:35 sdaj  66:48   active ready running
  |- 15:0:1:35 sdbx  68:176  active ready running
```

Step 4: Confirm the iSCSI configuration for your host

Ensure that iSCSI is configured correctly for your host.

About this task

You perform the following steps on the iSCSI host.

Steps

1. Verify that the iSCSI initiator package (iscsi-initiator-utils) is installed:

```
rpm -qa | grep iscsi-initiator-utils
```

   You should see an output similar to the following example:

```
iscsi-initiator-utils-6.2.1.11-0.git4b3e853.el9.x86_64
```

2. Verify the iSCSI initiator node name, which is located in the `/etc/iscsi/initiatorname.iscsi` file:

```
InitiatorName=iqn.YYYY-MM.com.<vendor>:<host_name>
```

3. Configure the iSCSI session timeout parameter located in the `/etc/iscsi/iscsid.conf` file:

```
node.session.timeo.replacement_timeout = 5
```

   The iSCSI `replacement_timeout` parameter controls how long the iSCSI layer should wait for a timed-out path or session to reestablish itself before failing any commands on it. You should set the value of `replacement_timeout` to 5 in the iSCSI configuration file.

4. Enable the iSCSI service:

```
$systemctl enable iscsid
```

5. Start the iSCSI service:

```
$systemctl start iscsid
```

6. Verify that the iSCSI service is running:

```
$systemctl status iscsid
```

Show example

```
● iscsid.service - Open-iSCSI
     Loaded: loaded (/usr/lib/systemd/system/iscsid.service;
enabled; preset: disabled)
     Active: active (running) since Tue 2025-12-02 11:36:21 EST; 2
weeks 1 day ago
TriggeredBy: ● iscsid.socket
       Docs: man:iscsid(8)
             man:iscsiuio(8)
             man:iscsiadm(8)
   Main PID: 2263 (iscsid)
     Status: "Ready to process requests"
      Tasks: 1 (limit: 816061)
     Memory: 18.5M
        CPU: 14.480s
     CGroup: /system.slice/iscsid.service
             └─2263 /usr/sbin/iscsid -f -d2
```

7. Discover the iSCSI targets:

```
$iscsiadm --mode discovery --op update --type sendtargets --portal
<target_IP>
```

show example

```
iscsiadm --mode discovery --op update --type sendtargets --portal
192.168.30.87
192.168.30.87:3260,1139 iqn.1992-
08.com.netapp:sn.064a9b19b3ee11f09dcad039eabac370:vs.23
192.168.31.97:3260,1142 iqn.1992-
08.com.netapp:sn.064a9b19b3ee11f09dcad039eabac370:vs.23
192.168.31.87:3260,1141 iqn.1992-
08.com.netapp:sn.064a9b19b3ee11f09dcad039eabac370:vs.23
192.168.30.97:3260,1140 iqn.1992-
08.com.netapp:sn.064a9b19b3ee11f09dcad039eabac370:vs.23
```

8. Log in to the targets:

```
$iscsiadm --mode node -l all
```

9. Set iSCSI to log in automatically when the host boots:

```
$iscsiadm --mode node -T <target_name> -p <ip:port> -o update -n
node.startup -v automatic
```

You should see an output similar to the following example:

```
iscsiadm --mode node -T iqn.1992-
08.com.netapp:sn.064a9b19b3ee11f09dcad039eabac370:vs.23 -p
192.168.30.87:3260 -o update -n node.startup -v automatic
```

10. Verify the iSCSI sessions:

```
$iscsiadm --mode session
```

Show example

```
iscsiadm --mode session
tcp: [1] 192.168.30.87:3260,1139 iqn.1992-
08.com.netapp:sn.064a9b19b3ee11f09dcad039eabac370:vs.23 (non-flash)
tcp: [2] 192.168.31.97:3260,1142 iqn.1992-
08.com.netapp:sn.064a9b19b3ee11f09dcad039eabac370:vs.23 (non-flash)
tcp: [3] 192.168.31.87:3260,1141 iqn.1992-
08.com.netapp:sn.064a9b19b3ee11f09dcad039eabac370:vs.23 (non-flash)
tcp: [4] 192.168.30.97:3260,1140 iqn.1992-
08.com.netapp:sn.064a9b19b3ee11f09dcad039eabac370:vs.23 (non-flash)
```

Step 5: Optionally, exclude a device from multipathing

If required, you can exclude a device from multipathing by adding the WWID for the unwanted device to the "blacklist" stanza for the `multipath.conf` file.

Steps

1. Determine the WWID:

```
/lib/udev/scsi_id -gud /dev/sda
```

"sda" is the local SCSI disk that you want to add to the blacklist.

An example WWID is `360030057024d0730239134810c0cb833`.

2. Add the WWID to the "blacklist" stanza:

```
blacklist {
         wwid   360030057024d0730239134810c0cb833
        devnode "^(ram|raw|loop|fd|md|dm-|sr|scd|st)[0-9]*"
        devnode "^hd[a-z]"
        devnode "^cciss.*"
}
```

Step 6: Customize multipath parameters for ONTAP LUNs

If your host is connected to LUNs from other vendors and any of the multipath parameter settings are overridden, you need to correct them by adding stanzas later in the `multipath.conf` file that apply specifically to ONTAP LUNs. If you don't do this, the ONTAP LUNs might not work as expected.

Check your `/etc/multipath.conf` file, especially in the defaults section, for settings that might be overriding the default settings for multipath parameters.

> You shouldn't override the recommended parameter settings for ONTAP LUNs. These settings are required for optimal performance of your host configuration. Contact NetApp support, your OS vendor, or both for more information.

The following example shows how to correct an overridden default. In this example, the `multipath.conf` file defines values for `path_checker` and `no_path_retry` that aren't compatible with ONTAP LUNs, and you can't remove these parameters because ONTAP storage arrays are still attached to the host. Instead, you correct the values for `path_checker` and `no_path_retry` by adding a device stanza to the `multipath.conf` file that applies specifically to the ONTAP LUNs.

Show example

```
defaults {
   path_checker       readsector0
   no_path_retry      fail
}

devices {
   device {
      vendor           "NETAPP"
      product          "LUN"
      no_path_retry    queue
      path_checker     tur
   }
}
```

Step 7: Review the known issues

There are no known issues.

For CentOS Red Hat compatible kernel known issues, see the known issues for the Red Hat Enterprise Linux (RHEL) 8.x series.

What's next?

- Learn about using the Linux Host Utilities tool.
- Learn about ASM mirroring

  Automatic Storage Management (ASM) mirroring might require changes to the Linux multipath settings to allow ASM to recognize a problem and switch over to an alternate failure group. Most ASM configurations on ONTAP use external redundancy, which means that data protection is provided by the external array and ASM doesn't mirror data. Some sites use ASM with normal redundancy to provide two-way mirroring, normally across different sites. See Oracle databases on ONTAP for further information.

- Learn about CentOS Linux Virtualization (KVM)

  CentOS Linux can serve as a KVM host. This enables you to run multiple virtual machines on a single physical server using the Linux Kernel-based Virtual Machine (KVM) technology. The KVM host doesn't require explicit host configuration settings for ONTAP LUNs.

Citrix

Configure Citrix Xenserver 8.4 for FCP and iSCSI with ONTAP storage

Configure Citrix Xenserver 8.4 for multipathing and with specific parameters and settings for FCP and iSCSI protocol operations with ONTAP storage.

The Linux Host Utilities software package doesn't support Citrix Xenserver operating systems.

Step 1: Optionally, enable SAN booting

You can configure your host to use SAN booting to simplify deployment and improve scalability.

Before you begin

Use the Interoperability Matrix Tool to verify that your Linux OS, host bus adapter (HBA), HBA firmware, HBA boot BIOS, and ONTAP version support SAN booting.

Steps

1. Create a SAN boot LUN and map it to the host.
2. Enable SAN booting in the server BIOS for the ports to which the SAN boot LUN is mapped.

   For information on how to enable the HBA BIOS, see your vendor-specific documentation.

3. Verify that the configuration was successful by rebooting the host and verifying that the OS is up and running.

Step 2: Confirm the multipath configuration for your host

You can use multipathing with Citrix Xenserver 8.4 to manage ONTAP LUNs.

To ensure that multipathing is configured correctly for your host, verify that the `/etc/multipath.conf` file is defined and that you have the NetApp recommended settings configured for your ONTAP LUNs.

Steps

1. Verify that the `/etc/multipath.conf` file exists. If the file doesn't exist, create an empty, zero-byte file:

```
touch /etc/multipath.conf
```

2. The first time the `multipath.conf` file is created, you might need to enable and start the multipath services to load the recommended settings:

```
systemctl enable multipathd
```

```
systemctl start multipathd
```

3. Each time you boot the host, the empty `/etc/multipath.conf` zero-byte file automatically loads the NetApp recommended host multipath parameters as the default settings. You shouldn't need to make changes to the `/etc/multipath.conf` file for your host because the host operating system is compiled with the multipath parameters that recognize and manage ONTAP LUNs correctly.

   The following table shows the native Linux OS compiled multipath parameter settings for ONTAP LUNs.

Show parameter settings

Parameter	Setting
detect_prio	yes
dev_loss_tmo	"infinity"
failback	immediate
fast_io_fail_tmo	5
features	"3 queue_if_no_path pg_init_retries 50"
flush_on_last_del	"yes"
hardware_handler	"0"
no_path_retry	queue
path_checker	"tur"
path_grouping_policy	"group_by_prio"
path_selector	"service-time 0"
polling_interval	5
prio	"ontap"
product	LUN
retain_attached_hw_handler	yes
rr_weight	"uniform"
user_friendly_names	no
vendor	NETAPP

4. Verify the parameter settings and path status for your ONTAP LUNs:

```
/sbin/mpathutil list
```

The default multipath parameters support ASA, AFF, and FAS configurations. In these configurations, a single ONTAP LUN shouldn’t require more than four paths. Having more than four paths can cause problems during a storage failure.

The following example outputs show the correct parameter settings and path status for ONTAP LUNs in an ASA, AFF, or FAS configuration.

ASA configuration

An ASA configuration optimizes all paths to a given LUN, keeping them active. This improves performance by serving I/O operations through all paths at the same time.

Show example

```
/usr/sbin/mpathutil status
show topology
create: 3600a098038315045572b5930646f4b63 dm-1 NETAPP  ,LUN C-
Mode
size=9.0G features='4 queue_if_no_path pg_init_retries 50
retain_attached_hw_handle' hwhandler='1 alua' wp=rw
`-+- policy='service-time 0' prio=50 status=active
  |- 15:0:0:0  sdb  8:16    active ready running
  |- 15:0:1:0  sdc  8:32    active ready running
  |- 16:0:0:0  sdcf 69:48   active ready running
  `- 16:0:1:0  sdcg 69:64   active ready running
```

AFF or FAS configuration

An AFF or FAS configuration should have two groups of paths with higher and lower priorities. Higher priority Active/Optimized paths are served by the controller where the aggregate is located. Lower priority paths are active but non-optimized because they are served by a different controller. Non-optimized paths are only used when optimized paths aren’t available.

The following example displays the output for an ONTAP LUN with two Active/Optimized paths and two Active/Non-Optimized paths:

Show example

```
/usr/sbin/mpathutil status
show topology
create: 3600a098038315045572b5930646f4b63 dm-1 NETAPP  ,LUN C-
Mode
size=9.0G features='4 queue_if_no_path pg_init_retries 50
retain_attached_hw_handle' hwhandler='1 alua' wp=rw
|-+- policy='service-time 0' prio=50 status=active
  |- 15:0:0:0  sdb  8:16    active ready running
  `- 15:0:1:0  sdc  8:32    active ready running
`-+- policy='service-time 0' prio=50 status=active
    |- 16:0:0:0  sdcf 69:48   active ready running
    `- 16:0:1:0  sdcg 69:64   active ready running
```

Step 3: Optionally, exclude a device from multipathing

If required, you can exclude a device from multipathing by adding the WWID for the unwanted device to the "blacklist" stanza for the `multipath.conf` file.

Steps

1. Determine the WWID:

```
/lib/udev/scsi_id -gud /dev/sda
```

   "sda" is the local SCSI disk that you want to add to the blacklist.

   An example WWID is `360030057024d0730239134810c0cb833`.

2. Add the WWID to the "blacklist" stanza:

```
blacklist {
         wwid   360030057024d0730239134810c0cb833
        devnode "^(ram|raw|loop|fd|md|dm-|sr|scd|st)[0-9]*"
        devnode "^hd[a-z]"
        devnode "^cciss.*"
}
```

Step 4: Customize multipath parameters for ONTAP LUNs

If your host is connected to LUNs from other vendors and any of the multipath parameter settings are overridden, you need to correct them by adding stanzas later in the `multipath.conf` file that apply specifically to ONTAP LUNs. If you don't do this, the ONTAP LUNs might not work as expected.

Check your `/etc/multipath.conf` file, especially in the defaults section, for settings that might be overriding the default settings for multipath parameters.

> You shouldn't override the recommended parameter settings for ONTAP LUNs. These settings are required for optimal performance of your host configuration. Contact NetApp support, your OS vendor, or both for more information.

The following example shows how to correct an overridden default. In this example, the `multipath.conf` file defines values for `path_checker` and `no_path_retry` that aren't compatible with ONTAP LUNs, and you can't remove these parameters because ONTAP storage arrays are still attached to the host. Instead, you correct the values for `path_checker` and `no_path_retry` by adding a device stanza to the `multipath.conf` file that applies specifically to the ONTAP LUNs.

Show example

```
defaults {
   path_checker       readsector0
   no_path_retry      fail
}

devices {
   device {
      vendor           "NETAPP"
      product          "LUN"
      no_path_retry    queue
      path_checker     tur
   }
}
```

Step 5: Review the known issues

There are no known issues.

Configure Citrix Hypervisor 8.2 for FCP and iSCSI with ONTAP storage

Configure Citrix Hypervisor 8.2 for multipathing and with specific parameters and settings for FCP and iSCSI protocol operations with ONTAP storage.

The Linux Host Utilities software package doesn't support Citrix Hypervisor operating systems.

Step 1: Optionally, enable SAN booting

You can configure your host to use SAN booting to simplify deployment and improve scalability.

Before you begin

Use the Interoperability Matrix Tool to verify that your Linux OS, host bus adapter (HBA), HBA firmware, HBA boot BIOS, and ONTAP version support SAN booting.

Steps

1. Create a SAN boot LUN and map it to the host.
2. Enable SAN booting in the server BIOS for the ports to which the SAN boot LUN is mapped.

   For information on how to enable the HBA BIOS, see your vendor-specific documentation.
3. Verify that the configuration was successful by rebooting the host and verifying that the OS is up and running.

Step 2: Confirm the multipath configuration for your host

You can use multipathing with Citrix Hypervisor 8.2 to manage ONTAP LUNs.

To ensure that multipathing is configured correctly for your host, verify that the `/etc/multipath.conf` file is defined and that you have the NetApp recommended settings configured for your ONTAP LUNs.

Steps

1. Verify that the `/etc/multipath.conf` file exists. If the file doesn't exist, create an empty, zero-byte file:

```
touch /etc/multipath.conf
```

2. The first time the `multipath.conf` file is created, you might need to enable and start the multipath services to load the recommended settings:

```
systemctl enable multipathd
```

```
systemctl start multipathd
```

3. Each time you boot the host, the empty `/etc/multipath.conf` zero-byte file automatically loads the NetApp recommended host multipath parameters as the default settings. You shouldn't need to make changes to the `/etc/multipath.conf` file for your host because the host operating system is compiled with the multipath parameters that recognize and manage ONTAP LUNs correctly.

   The following table shows the native Linux OS compiled multipath parameter settings for ONTAP LUNs.

Show parameter settings

Parameter	Setting
detect_prio	yes
dev_loss_tmo	"infinity"
failback	immediate
fast_io_fail_tmo	5
features	"3 queue_if_no_path pg_init_retries 50"
flush_on_last_del	"yes"
hardware_handler	"0"
no_path_retry	queue
path_checker	"tur"
path_grouping_policy	"group_by_prio"
path_selector	"service-time 0"
polling_interval	5
prio	"ontap"
product	LUN
retain_attached_hw_handler	yes
rr_weight	"uniform"
user_friendly_names	no
vendor	NETAPP

4. Verify the parameter settings and path status for your ONTAP LUNs:

```
/sbin/mpathutil list
```

The default multipath parameters support ASA, AFF, and FAS configurations. In these configurations, a single ONTAP LUN shouldn’t require more than four paths. Having more than four paths can cause problems during a storage failure.

The following example outputs show the correct parameter settings and path status for ONTAP LUNs in an ASA, AFF, or FAS configuration.

ASA configuration

An ASA configuration optimizes all paths to a given LUN, keeping them active. This improves performance by serving I/O operations through all paths at the same time.

Show example

```
/usr/sbin/mpathutil status
show topology
create: 3600a098038315045572b5930646f4b63 dm-1 NETAPP  ,LUN C-
Mode
size=9.0G features='4 queue_if_no_path pg_init_retries 50
retain_attached_hw_handle' hwhandler='1 alua' wp=rw
`-+- policy='service-time 0' prio=50 status=active
  |- 15:0:0:0  sdb  8:16    active ready running
  |- 15:0:1:0  sdc  8:32    active ready running
  |- 16:0:0:0  sdcf 69:48   active ready running
  `- 16:0:1:0  sdcg 69:64   active ready running
```

AFF or FAS configuration

An AFF or FAS configuration should have two groups of paths with higher and lower priorities. Higher priority Active/Optimized paths are served by the controller where the aggregate is located. Lower priority paths are active but non-optimized because they are served by a different controller. Non-optimized paths are only used when optimized paths aren’t available.

The following example displays the output for an ONTAP LUN with two Active/Optimized paths and two Active/Non-Optimized paths:

Show example

```
/usr/sbin/mpathutil status
show topology
create: 3600a098038315045572b5930646f4b63 dm-1 NETAPP  ,LUN C-
Mode
size=9.0G features='4 queue_if_no_path pg_init_retries 50
retain_attached_hw_handle' hwhandler='1 alua' wp=rw
|-+- policy='service-time 0' prio=50 status=active
  |- 15:0:0:0  sdb  8:16    active ready running
  `- 15:0:1:0  sdc  8:32    active ready running
`-+- policy='service-time 0' prio=50 status=active
    |- 16:0:0:0  sdcf 69:48   active ready running
    `- 16:0:1:0  sdcg 69:64   active ready running
```

Step 3: Optionally, exclude a device from multipathing

If required, you can exclude a device from multipathing by adding the WWID for the unwanted device to the "blacklist" stanza for the `multipath.conf` file.

Steps

1. Determine the WWID:

```
/lib/udev/scsi_id -gud /dev/sda
```

   "sda" is the local SCSI disk that you want to add to the blacklist.

   An example WWID is `360030057024d0730239134810c0cb833`.

2. Add the WWID to the "blacklist" stanza:

```
blacklist {
         wwid   360030057024d0730239134810c0cb833
        devnode "^(ram|raw|loop|fd|md|dm-|sr|scd|st)[0-9]*"
        devnode "^hd[a-z]"
        devnode "^cciss.*"
}
```

Step 4: Customize multipath parameters for ONTAP LUNs

If your host is connected to LUNs from other vendors and any of the multipath parameter settings are overridden, you need to correct them by adding stanzas later in the `multipath.conf` file that apply specifically to ONTAP LUNs. If you don't do this, the ONTAP LUNs might not work as expected.

Check your `/etc/multipath.conf` file, especially in the defaults section, for settings that might be overriding the default settings for multipath parameters.

> You shouldn't override the recommended parameter settings for ONTAP LUNs. These settings are required for optimal performance of your host configuration. Contact NetApp support, your OS vendor, or both for more information.

The following example shows how to correct an overridden default. In this example, the `multipath.conf` file defines values for `path_checker` and `no_path_retry` that aren't compatible with ONTAP LUNs, and you can't remove these parameters because ONTAP storage arrays are still attached to the host. Instead, you correct the values for `path_checker` and `no_path_retry` by adding a device stanza to the `multipath.conf` file that applies specifically to the ONTAP LUNs.

Show example

```
defaults {
   path_checker       readsector0
   no_path_retry      fail
}

devices {
   device {
      vendor           "NETAPP"
      product          "LUN"
      no_path_retry    queue
      path_checker     tur
   }
}
```

Step 5: Review the known issues

There are no known issues.

ESXi

Use VMware vSphere 8.x with ONTAP

You can configure ONTAP SAN host settings for the VMware vSphere 8.x release with FC, FCoE, and iSCSI protocols.

Hypervisor SAN booting

Before you begin

If you decide to use SAN booting, it must be supported by your configuration. You can use the Interoperability Matrix Tool to verify that your OS, HBA, HBA firmware and the HBA boot BIOS, and ONTAP version are supported.

Steps

1. Map the SAN boot LUN to the host.
2. Verify that multiple paths are available.

   Multiple paths become available after the host OS is up and running on the paths.

3. Enable SAN booting in the server BIOS for the ports to which the SAN boot LUN is mapped.

   For information on how to enable the HBA BIOS, see your vendor-specific documentation.

4. Reboot the host to verify that the boot was successful.

Multipathing

ESXi provides an extensible multipathing module called Native Multipathing Plug-In (NMP) that manages the sub-plugins, Storage Array Type Plugins (SATPs), and Path Selection Plugins (PSPs). By default, these SATP rules are available in ESXi.

For ONTAP storage, the "VMW_SATP_ALUA" plugin is used by default with "VMW_PSP_RR" as a path selection policy (PSP). You can run the following command to confirm the PSP:

```
`esxcli storage nmp satp rule list -s VMW_SATP_ALUA`
```

Example output:

```
Name           Device  Vendor    Model         Driver  Transport  Options
-------------------------------------------------------------------------
-------
VMW_SATP_ALUA          LSI       INF-01-00
reset_on_attempted_reserve  system
VMW_SATP_ALUA          NETAPP
reset_on_attempted_reserve  system

Rule Group  Claim Options  Default PSP  PSP Options     Description
----------------------------------------------------------------
tpgs_on        VMW_PSP_MRU                 NetApp E-Series arrays with
ALUA support
tpgs_on        VMW_PSP_RR                  NetApp arrays with ALUA
support
```

Non-ASA configurations

For non-ASA configurations, there should be two groups of paths with different priorities. The paths with higher priorities are Active/Optimized. This means they are serviced by the controller where the aggregate is located. The paths with lower priorities are active but non-optimized because they are served from a different controller. The non-optimized paths are only used when optimized paths are not available.

Example

The following example displays the correct output for an ONTAP LUN with two Active/Optimized paths and two Active/Non-Optimized paths.

```
# esxcli storage nmp device list -d naa.600a0980383148693724545244395855
```

Example output:

```
naa.600a0980383148693724545244395855
   Device Display Name: NETAPP Fibre Channel Disk
(naa.600a0980383148693724545244395855)
   Storage Array Type: VMW_SATP_ALUA
   Storage Array Type Device Config: {implicit_support=on;
explicit_support=off; explicit_allow=on; alua_followover=on;
action_OnRetryErrors=off;
{TPG_id=1000,TPG_state=ANO}{TPG_id=1001,TPG_state=AO}}
   Path Selection Policy: VMW_PSP_RR
   Path Selection Policy Device Config:
{policy=rr,iops=1000,bytes=10485760,useANO=0; lastPathIndex=1:
NumIOsPending=0,numBytesPending=0}
   Path Selection Policy Device Custom Config:
   Working Paths: vmhba4:C0:T0:L11, vmhba3:C0:T0:L11
   Is USB: false
```

```
# esxcli storage nmp path list -d naa.600a0980383148693724545244395855
```

Example output:

```
fc.20000024ff7f4a51:21000024ff7f4a51-fc.2009d039ea3ab21f:2003d039ea3ab21f-
naa.600a0980383148693724545244395855
   Runtime Name: vmhba4:C0:T0:L11
   Device: naa.600a0980383148693724545244395855
   Device Display Name: NETAPP Fibre Channel Disk
(naa.600a0980383148693724545244395855)
   Group State: active
   Array Priority: 0
   Storage Array Type Path Config: {TPG_id=1001,
TPG_state=AO,RTP_id=4,RTP_health=UP}
   Path Selection Policy Path Config: PSP VMW_PSP_RR does not support path
configuration.

fc.20000024ff7f4a50:21000024ff7f4a50-fc.2009d039ea3ab21f:2002d039ea3ab21f-
naa.600a0980383148693724545244395855
   Runtime Name: vmhba3:C0:T0:L11
   Device: naa.600a0980383148693724545244395855
   Device Display Name: NETAPP Fibre Channel Disk
(naa.600a0980383148693724545244395855)
   Group State: active
   Array Priority: 0
   Storage Array Type Path Config: {TPG_id=1001,
TPG_state=AO,RTP_id=3,RTP_health=UP}
```

```
   Path Selection Policy Path Config: PSP VMW_PSP_RR does not support path
configuration.

fc.20000024ff7f4a51:21000024ff7f4a51-fc.2009d039ea3ab21f:2001d039ea3ab21f-
naa.600a098038314869372454524395855
   Runtime Name: vmhba4:C0:T3:L11
   Device: naa.600a098038314869372454524395855
   Device Display Name: NETAPP Fibre Channel Disk
(naa.600a098038314869372454524395855)
   Group State: active unoptimized
   Array Priority: 0
   Storage Array Type Path Config: {TPG_id=1000,
TPG_state=ANO,RTP_id=2,RTP_health=UP}
   Path Selection Policy Path Config: PSP VMW_PSP_RR does not support path
configuration.

fc.20000024ff7f4a50:21000024ff7f4a50-fc.2009d039ea3ab21f:2000d039ea3ab21f-
naa.600a098038314869372454524395855
   Runtime Name: vmhba3:C0:T3:L11
   Device: naa.600a098038314869372454524395855
   Device Display Name: NETAPP Fibre Channel Disk
(naa.600a098038314869372454524395855)
   Group State: active unoptimized
   Array Priority: 0
   Storage Array Type Path Config: {TPG_id=1000,
TPG_state=ANO,RTP_id=1,RTP_health=UP}
   Path Selection Policy Path Config: PSP VMW_PSP_RR does not support path
configuration.
```

Unresolved directive in hu_vsphere_8.adoc - include::_include/hu/reuse_hu_asa_configuration.adoc[]

```
esxcli storage nmp device list -d naa.600a098038304759563f4e7837574453
```

Example output:

```
naa.600a098038314962485d543078486c7a
   Device Display Name: NETAPP Fibre Channel Disk
(naa.600a098038314962485d543078486c7a)
   Storage Array Type: VMW_SATP_ALUA
   Storage Array Type Device Config: {implicit_support=on;
explicit_support=off; explicit_allow=on; alua_followover=on;
action_OnRetryErrors=off;
{TPG_id=1001,TPG_state=AO}{TPG_id=1000,TPG_state=AO}}
   Path Selection Policy: VMW_PSP_RR
   Path Selection Policy Device Config:
{policy=rr,iops=1000,bytes=10485760,useANO=0; lastPathIndex=3:
NumIOsPending=0,numBytesPending=0}
   Path Selection Policy Device Custom Config:
   Working Paths: vmhba4:C0:T0:L14, vmhba4:C0:T1:L14, vmhba3:C0:T0:L14,
vmhba3:C0:T1:L14
   Is USB: false
```

```
# esxcli storage nmp path list -d naa.600a098038314962485d543078486c7a
```

Example output:

```
fc.200034800d756a75:210034800d756a75-fc.2018d039ea936319:2015d039ea936319-
naa.600a098038314962485d543078486c7a
   Runtime Name: vmhba4:C0:T0:L14
   Device: naa.600a098038314962485d543078486c7a
   Device Display Name: NETAPP Fibre Channel Disk
(naa.600a098038314962485d543078486c7a)
   Group State: active
   Array Priority: 0
   Storage Array Type Path Config: {TPG_id=1000,
TPG_state=AO,RTP_id=2,RTP_health=UP}
   Path Selection Policy Path Config: PSP VMW_PSP_RR does not support path
configuration.

fc.200034800d756a75:210034800d756a75-fc.2018d039ea936319:2017d039ea936319-
naa.600a098038314962485d543078486c7a
   Runtime Name: vmhba4:C0:T1:L14
   Device: naa.600a098038314962485d543078486c7a
   Device Display Name: NETAPP Fibre Channel Disk
(naa.600a098038314962485d543078486c7a)
   Group State: active
   Array Priority: 0
   Storage Array Type Path Config: {TPG_id=1001,
```

```
TPG_state=AO,RTP_id=4,RTP_health=UP}
   Path Selection Policy Path Config: PSP VMW_PSP_RR does not support path
configuration.

fc.200034800d756a74:210034800d756a74-fc.2018d039ea936319:2014d039ea936319-
naa.600a098038314962485d543078486c7a
   Runtime Name: vmhba3:C0:T0:L14
   Device: naa.600a098038314962485d543078486c7a
   Device Display Name: NETAPP Fibre Channel Disk
(naa.600a098038314962485d543078486c7a)
   Group State: active
   Array Priority: 0
   Storage Array Type Path Config: {TPG_id=1000,
TPG_state=AO,RTP_id=1,RTP_health=UP}
   Path Selection Policy Path Config: PSP VMW_PSP_RR does not support path
configuration.

fc.200034800d756a74:210034800d756a74-fc.2018d039ea936319:2016d039ea936319-
naa.600a098038314962485d543078486c7a
   Runtime Name: vmhba3:C0:T1:L14
   Device: naa.600a098038314962485d543078486c7a
   Device Display Name: NETAPP Fibre Channel Disk
(naa.600a098038314962485d543078486c7a)
   Group State: active
   Array Priority: 0
   Storage Array Type Path Config: {TPG_id=1001,
TPG_state=AO,RTP_id=3,RTP_health=UP}
   Path Selection Policy Path Config: PSP VMW_PSP_RR does not support path
configuration.
```

vVol

Virtual Volumes (vVols) are a VMware object type that corresponds to a Virtual Machine (VM) disk, its snapshots, and fast clones.

ONTAP tools for VMware vSphere includes the VASA Provider for ONTAP, which provides the integration point for a VMware vCenter to leverage vVols based storage. When you deploy the ONTAP tools Open Virtualization Appliance (OVA), it is automatically registered with the vCenter server and enables the VASA Provider.

When you create a vVols datastore using the vCenter UI, it guides you to create FlexVols as backup storage for the datastore. vVols within vVols datastores are accessed by ESXi hosts using a protocol endpoint (PE). In SAN environments, one 4MB LUN is created on each FlexVol in the datastore for use as a PE. A SAN PE is an administrative logical unit (ALU). vVols are subsidiary logical units (SLUs).

Standard requirements and best practices for SAN environments apply when using vVols, including (but not limited to) the following:

- Create at least one SAN LIF on each node per SVM you intend to use. The best practice is to create at least two per node, but no more than necessary.

- Eliminate any single point of failure. Use multiple VMkernel network interfaces on different network subnets that use NIC teaming when multiple virtual switches are used, or use multiple physical NICs connected to multiple physical switches to provide HA and increased throughput.
- Configure zoning, VLANs, or both as required for host connectivity.
- Verify that all required initiators are logged into the target LIFs on the desired SVM.

> You must deploy ONTAP tools for VMware vSphere to enable the VASA Provider. The VASA Provider will manage all of your iGroup settings for you, therefore there is no need to create or manage iGroups in a vVols environment.

NetApp does not recommend changing any vVols settings from default at this time.

Refer to the Interoperability Matrix Tool for specific versions of ONTAP tools, or legacy VASA Provider for your specific versions of vSphere and ONTAP.

For detailed information on provisioning and managing vVols, refer to the ONTAP tools for VMware vSphere documentation, VMware vSphere with ONTAP, and Virtual Volumes (vVols) with ONTAP tools 10.

Recommended settings

ATS locking

ATS locking is **mandatory** for VAAI compatible storage and upgraded VMFS5 and is required for proper interoperability and optimal VMFS shared storage I/O performance with ONTAP LUNs. Refer to VMware documentation for details on enabling ATS locking.

Settings	Default	ONTAP Recommended	Description
HardwareAcceleratedLocking	1	1	Helps enable the use of Atomic Test and Set (ATS) locking
Disk IOPs	1000	1	IOPS limit: The Round Robin PSP defaults to an IOPS limit of 1000. In this default case, a new path is used after 1000 I/O operations are issued.
Disk/QFullSampleSize	0	32	The count of QUEUE FULL or BUSY conditions it takes before ESXi starts throttling.

> Enable `Space-alloc` setting for all the LUNs mapped to VMware vSphere for `UNMAP` to work. For more details, refer to ONTAP Documentation.

Guest OS timeouts

You can manually configure the virtual machines with the recommended guest OS tunings. After tuning updates, you must reboot the guest for the updates to take effect.

GOS timeout values:

Guest OS Type	Timeouts
Linux variants	disk timeout = 60
Windows	disk timeout = 60
Solaris	disk timeout = 60 busy retry = 300 not ready retry = 300 reset retry = 30 max.throttle = 32 min.throttle = 8

Validate the vSphere tunable

You can use the following command to verify the `HardwareAcceleratedLocking` setting.

```
esxcli system settings advanced list --option /VMFS3/HardwareAcceleratedLocking
```

```
    Path: /VMFS3/HardwareAcceleratedLocking
    Type: integer
    Int Value: 1
    Default Int Value: 1
    Min Value: 0
    Max Value: 1
    String Value:
    Default String Value:
    Valid Characters:
    Description: Enable hardware accelerated VMFS locking (requires
compliant hardware). Please see http://kb.vmware.com/kb/2094604 before
disabling this option.
```

Validate the disk IOPs setting

You can use the following command to verify the IOPs setting.

```
esxcli storage nmp device list -d naa.600a098038304731783f506670553355
```

```
naa.600a0980383047317783f506670553355
   Device Display Name: NETAPP Fibre Channel Disk
(naa.600a0980383047317783f506670553355)
   Storage Array Type: VMW_SATP_ALUA
   Storage Array Type Device Config: {implicit_support=on;
explicit_support=off; explicit_allow=on; alua_followover=on;
action_OnRetryErrors=off;
{TPG_id=1000,TPG_state=ANO}{TPG_id=1001,TPG_state=AO}}
   Path Selection Policy: VMW_PSP_RR
   Path Selection Policy Device Config: {policy=rr,
iops=1,bytes=10485760,useANO=0; lastPathIndex=0:
NumIOsPending=0,numBytesPending=0}
   Path Selection Policy Device Custom Config:
   Working Paths: vmhba4:C0:T0:L82, vmhba3:C0:T0:L82
   Is USB: false
```

Validate the QFullSampleSize

You can use the following command to verify the QFullSampleSize.

```
esxcli system settings advanced list --option /Disk/QFullSampleSize
```

```
   Path: /Disk/QFullSampleSize
   Type: integer
   Int Value: 32
   Default Int Value: 0
   Min Value: 0
   Max Value: 64
   String Value:
   Default String Value:
   Valid Characters:
   Description: Default I/O samples to monitor for detecting non-transient
queue full condition. Should be nonzero to enable queue depth throttling.
Device specific QFull options will take precedence over this value if set.
```

Known issues

The VMware vSphere 8.x with ONTAP release has the following known issues:

NetApp Bug ID	Title	Description
1543660	I/O error occurs when Linux VMs using vNVMe adapters encounter a long all paths down (APD) window	Linux VMs running vSphere 8.x and later and using virtual NVMe (vNVME) adapters encounter an I/O error because the vNVMe retry operation is disabled by default. To avoid a disruption on Linux VMs running older kernels during an all paths down (APD) or a heavy I/O load, VMware has introduced a tunable "VSCSIDisableNvmeRetry" to disable the vNVMe retry operation.

Related information

- VMware vSphere with ONTAP
- VMware vSphere 5.x, 6.x and 7.x support with NetApp MetroCluster (2031038)
- NetApp ONTAP with NetApp SnapMirror active sync with VMware vSphere Metro Storage Cluster (vMSC)

Use VMware vSphere 7.x with ONTAP

You can use ONTAP SAN host configuration settings for the vSphere 7.x release with FC, FCoE and iSCSI protocols.

Hypervisor SAN Booting

Before you begin

If you decide to use SAN booting, it must be supported by your configuration. You can use the Interoperability Matrix Tool to verify that your OS, HBA, HBA firmware and the HBA boot BIOS, and ONTAP version are supported.

Steps

1. Map the SAN boot LUN to the host.
2. Verify that multiple paths are available.

   Multiple paths become available after the host OS is up and running on the paths.

3. Enable SAN booting in the server BIOS for the ports to which the SAN boot LUN is mapped.

   For information on how to enable the HBA BIOS, see your vendor-specific documentation.

4. Reboot the host to verify that the boot was successful.

Multipathing

ESXi provides an extensible multipathing module called Native Multipathing Plug-In (NMP) that manages the sub-plugins, Storage Array Type Plugins (SATPs), and Path Selection Plugins (PSPs). These SATP rules are available by default in ESXi.

For ONTAP storage, the "VMW_SATP_ALUA" plugin is used by default with "VMW_PSP_RR" as a path selection policy (PSP). You can run the following command to confirm the PSP.

esxcli storage nmp satp rule list -s VMW_SATP_ALUA

```
Name           Device  Vendor    Model         Driver  Transport  Options
-------------  ------  --------  ------------  ------  ---------
--------------------------
VMW_SATP_ALUA          NETAPP
reset_on_attempted_reserve


Rule Group  Claim Options  Default PSP  PSP Options     Description
 ----------  -------------  -----------  --------------  -----------
system      tpgs_on        VMW_PSP_RR                   NetApp arrays with
ALUA support
```

Non-ASA configurations

For non-ASA configurations, there should be two groups of paths with different priorities. The paths with higher priorities are Active/Optimized. This means they are serviced by the controller where the aggregate is located. The paths with lower priorities are active but non-optimized because they are served from a different controller. The non-optimized paths are only used when optimized paths are not available.

Example

The following example displays the correct output for an ONTAP LUN with two Active/Optimized paths and two Active/Non-Optimized paths.

esxcli storage nmp device list -d naa.600a098038313530772b4d673979372f

```
naa.600a098038313530772b4d673979372f
   Device Display Name: NETAPP Fibre Channel Disk
(naa.600a098038313530772b4d673979372f)
   Storage Array Type: VMW_SATP_ALUA
   Storage Array Type Device Config: {implicit_support=on;
explicit_support=off; explicit_allow=on; alua_followover=on;
action_OnRetryErrors=off;
{TPG_id=1000,TPG_state=AO}{TPG_id=1001,TPG_state=ANO}}
   Path Selection Policy: VMW_PSP_RR
   Path Selection Policy Device Config:
{policy=rr,iops=1,bytes=10485760,useANO=0; lastPathIndex=1:
NumIOsPending=0,numBytesPending=0}
   Path Selection Policy Device Custom Config:
   Working Paths: vmhba3:C0:T3:L21, vmhba4:C0:T2:L21
   Is USB: false
```

esxcli storage nmp path list -d naa.600a098038313530772b4d673979372f

```

```

```
fc.20000090fae0ec8e:10000090fae0ec8e-fc.201000a098dfe3d1:200b00a098dfe3d1-
naa.600a098038313530772b4d673979372f
   Runtime Name: vmhba3:C0:T2:L21
   Device: naa.600a098038313530772b4d673979372f
   Device Display Name: NETAPP Fibre Channel Disk
(naa.600a098038313530772b4d673979372f)
   Group State: active unoptimized
   Array Priority: 0
   Storage Array Type Path Config:
{TPG_id=1001,TPG_state=ANO,RTP_id=29,RTP_health=UP}
   Path Selection Policy Path Config: PSP VMW_PSP_RR does not support path
configuration.

fc.20000090fae0ec8e:10000090fae0ec8e-fc.201000a098dfe3d1:200700a098dfe3d1-
naa.600a098038313530772b4d673979372f
   Runtime Name: vmhba3:C0:T3:L21
   Device: naa.600a098038313530772b4d673979372f
   Device Display Name: NETAPP Fibre Channel Disk
(naa.600a098038313530772b4d673979372f)
   Group State: active
   Array Priority: 0
   Storage Array Type Path Config:
{TPG_id=1000,TPG_state=AO,RTP_id=25,RTP_health=UP}
   Path Selection Policy Path Config: PSP VMW_PSP_RR does not support path
configuration.

fc.20000090fae0ec8f:10000090fae0ec8f-fc.201000a098dfe3d1:200800a098dfe3d1-
naa.600a098038313530772b4d673979372f
   Runtime Name: vmhba4:C0:T2:L21
   Device: naa.600a098038313530772b4d673979372f
   Device Display Name: NETAPP Fibre Channel Disk
(naa.600a098038313530772b4d673979372f)
   Group State: active
   Array Priority: 0
   Storage Array Type Path Config:
{TPG_id=1000,TPG_state=AO,RTP_id=26,RTP_health=UP}
   Path Selection Policy Path Config: PSP VMW_PSP_RR does not support path
configuration.

fc.20000090fae0ec8f:10000090fae0ec8f-fc.201000a098dfe3d1:200c00a098dfe3d1-
naa.600a098038313530772b4d673979372f
   Runtime Name: vmhba4:C0:T3:L21
   Device: naa.600a098038313530772b4d673979372f
   Device Display Name: NETAPP Fibre Channel Disk
(naa.600a098038313530772b4d673979372f)
   Group State: active unoptimized
```

```
   Array Priority: 0
   Storage Array Type Path Config:
{TPG_id=1001,TPG_state=ANO,RTP_id=30,RTP_health=UP}
   Path Selection Policy Path Config: PSP VMW_PSP_RR does not support path
configuration.
```

Unresolved directive in hu_vsphere_7.adoc - include::_include/hu/reuse_hu_asa_configuration.adoc[]

esxcli storage nmp device list -d naa.600a098038304759563f4e7837574453

```
naa.600a098038304759563f4e7837574453
   Device Display Name: NETAPP Fibre Channel Disk
(naa.600a098038304759563f4e7837574453)
   Storage Array Type: VMW_SATP_ALUA
   Storage Array Type Device Config: {implicit_support=on;
explicit_support=off; explicit_allow=on; alua_followover=on;
action_OnRetryErrors=off;
{TPG_id=1001,TPG_state=AO}{TPG_id=1000,TPG_state=AO}}
   Path Selection Policy: VMW_PSP_RR
   Path Selection Policy Device Config:
{policy=rr,iops=1,bytes=10485760,useANO=0; lastPathIndex=2:
NumIOsPending=0,numBytesPending=0}
   Path Selection Policy Device Custom Config:
   Working Paths: vmhba4:C0:T0:L9, vmhba3:C0:T1:L9, vmhba3:C0:T0:L9,
vmhba4:C0:T1:L9
   Is USB: false
```

esxcli storage nmp device list -d naa.600a098038304759563f4e7837574453

```
fc.20000024ff171d37:21000024ff171d37-fc.202300a098ea5e27:204a00a098ea5e27-
naa.600a098038304759563f4e7837574453
   Runtime Name: vmhba4:C0:T0:L9
   Device: naa.600a098038304759563f4e7837574453
   Device Display Name: NETAPP Fibre Channel Disk
(naa.600a098038304759563f4e7837574453)
   Group State: active
   Array Priority: 0
   Storage Array Type Path Config:
{TPG_id=1000,TPG_state=AO,RTP_id=6,RTP_health=UP}
   Path Selection Policy Path Config: PSP VMW_PSP_RR does not support path
configuration.

fc.20000024ff171d36:21000024ff171d36-fc.202300a098ea5e27:201d00a098ea5e27-
naa.600a098038304759563f4e7837574453
   Runtime Name: vmhba3:C0:T1:L9
```

```
   Device: naa.600a098038304759563f4e7837574453
   Device Display Name: NETAPP Fibre Channel Disk
(naa.600a098038304759563f4e7837574453)
   Group State: active
   Array Priority: 0
   Storage Array Type Path Config:
{TPG_id=1001,TPG_state=AO,RTP_id=3,RTP_health=UP}
   Path Selection Policy Path Config: PSP VMW_PSP_RR does not support path
configuration.

fc.20000024ff171d36:21000024ff171d36-fc.202300a098ea5e27:201b00a098ea5e27-
naa.600a098038304759563f4e7837574453
   Runtime Name: vmhba3:C0:T0:L9
   Device: naa.600a098038304759563f4e7837574453
   Device Display Name: NETAPP Fibre Channel Disk
(naa.600a098038304759563f4e7837574453)
   Group State: active
   Array Priority: 0
   Storage Array Type Path Config:
{TPG_id=1000,TPG_state=AO,RTP_id=1,RTP_health=UP}
   Path Selection Policy Path Config: PSP VMW_PSP_RR does not support path
configuration.

fc.20000024ff171d37:21000024ff171d37-fc.202300a098ea5e27:201e00a098ea5e27-
naa.600a098038304759563f4e7837574453
   Runtime Name: vmhba4:C0:T1:L9
   Device: naa.600a098038304759563f4e7837574453
   Device Display Name: NETAPP Fibre Channel Disk
(naa.600a098038304759563f4e7837574453)
   Group State: active
   Array Priority: 0
   Storage Array Type Path Config:
{TPG_id=1001,TPG_state=AO,RTP_id=4,RTP_health=UP}
   Path Selection Policy Path Config: PSP VMW_PSP_RR does not support path
configuration.
```

vVol

Virtual Volumes (vVols) are a VMware object type that corresponds to a Virtual Machine (VM) disk, and its snapshots and fast-clones.

ONTAP tools for VMware vSphere includes the VASA Provider for ONTAP, which provides the integration point for a VMware vCenter to leverage vVols based storage. When you deploy the ONTAP tools OVA, it is automatically registered with the vCenter server and enables the VASA Provider.

When you create a vVols datastore using the vCenter UI, it guides you to create FlexVols as backup storage for the datastore. vVols within a vVols datastores are access by ESXi hosts using a protocol endpoint (PE). In SAN environments, one 4MB LUN is created on each FlexVol in the datastore for use as a PE. A SAN PE is an

administrative logical unit (ALU). vVols are subsidiary logical units (SLUs).

Standard requirements and best practices for SAN environments apply when using vVols, including (but not limited to) the following:

1. Create at least one SAN LIF on each node per SVM you intend to use. The best practice is to create at least two per node, but no more than necessary.
2. Eliminate any single point of failure. use multiple VMkernel network interfaces on different network subnets that use NIC teaming when multiple virtual switches are used. Or use multiple physical NICs connected to multiple physical switches to provide HA and increased throughput.
3. Configure zoning and/or VLANs as required for host connectivity.
4. Ensure all required initiators are logged into the target LIFs on the desired SVM.

You must deploy ONTAP tools for VMware vSphere to enabled the VASA Provider. The VASA Provider will manage all of your igroup settings for you, so there is no need to create or manage igroups in a vVols environment.

NetApp does not recommend changing any vVols settings from the default at this time.

Refer to the Interoperability Matrix Tool for specific versions of ONTAP tools, or legacy VASA Provider for your specific versions of vSphere and ONTAP.

For detailed information on provisioning and managing vVols, refer to the ONTAP tools for VMware vSphere documentation as well as VMware vSphere with ONTAP and Virtual Volumes (vVols) with ONTAP tools 10.

Recommended Settings

ATS Locking

ATS locking is **mandatory** for VAAI compatible storage and upgraded VMFS5, and is required for proper interoperability and optimal VMFS shared storage I/O performance with ONTAP LUNs. Refer to VMware documentation for details on enabling ATS locking.

Settings	Default	ONTAP Recommended	Description
HardwareAcceleratedLocking	1	1	Helps enable the use of Atomic Test and Set (ATS) locking
Disk IOPs	1000	1	IOPS limit: The Round Robin PSP defaults to an IOPS limit of 1000. In this default case, a new path is used after 1000 I/O operations are issued.
Disk/QFullSampleSize	0	32	The count of QUEUE FULL or BUSY conditions it takes before ESXi starts throttling.

Enable Space-alloc setting for all the LUN's mapped to VMware vSphere for UNMAP to work. For More details, refer to ONTAP Documentation.

Guest OS timeouts

You can manually configure the virtual machines with the recommended guest OS tunings. After tuning updates, you must reboot the guest for the updates to take effect.

GOS timeout values:

Guest OS Type	Timeouts
Linux variants	disk timeout = 60
Windows	disk timeout = 60
Solaris	disk timeout = 60 busy retry = 300 not ready retry = 300 reset retry = 30 max.throttle = 32 min.throttle = 8

Validating the vSphere tunable

Use the following command to verify the HardwareAcceleratedLocking setting.

esxcli system settings advanced list --option /VMFS3/HardwareAcceleratedLocking

```
   Path: /VMFS3/HardwareAcceleratedLocking
   Type: integer
   Int Value: 1
   Default Int Value: 1
   Min Value: 0
   Max Value: 1
   String Value:
   Default String Value:
   Valid Characters:
   Description: Enable hardware accelerated VMFS locking (requires
compliant hardware). Please see http://kb.vmware.com/kb/2094604 before
disabling this option.
```

Validate the disk IOPs setting

Use the following command to verify the IOPs setting.

esxcli storage nmp device list -d naa.600a098038304731783f506670553355

```
naa.600a0980383047317783f506670553355
   Device Display Name: NETAPP Fibre Channel Disk
(naa.600a0980383047317783f506670553355)
   Storage Array Type: VMW_SATP_ALUA
   Storage Array Type Device Config: {implicit_support=on;
explicit_support=off; explicit_allow=on; alua_followover=on;
action_OnRetryErrors=off;
{TPG_id=1000,TPG_state=ANO}{TPG_id=1001,TPG_state=AO}}
   Path Selection Policy: VMW_PSP_RR
   Path Selection Policy Device Config:
{policy=rr,iops=1,bytes=10485760,useANO=0; lastPathIndex=0:
NumIOsPending=0,numBytesPending=0}
   Path Selection Policy Device Custom Config:
   Working Paths: vmhba4:C0:T0:L82, vmhba3:C0:T0:L82
   Is USB: false
```

Validating the QFullSampleSize

Use the following command to verify the QFullSampleSize

`esxcli system settings advanced list --option /Disk/QFullSampleSize`

```
   Path: /Disk/QFullSampleSize
   Type: integer
   Int Value: 32
   Default Int Value: 0
   Min Value: 0
   Max Value: 64
   String Value:
   Default String Value:
   Valid Characters:
   Description: Default I/O samples to monitor for detecting non-transient
queue full condition. Should be nonzero to enable queue depth throttling.
Device specific QFull options will take precedence over this value if set.
```

Known issues

There are no known issues for the VMware vSphere 7.x with ONTAP release.

Related information

- VMware vSphere with ONTAP
- VMware vSphere 5.x, 6.x and 7.x support with NetApp MetroCluster (2031038)
- NetApp ONTAP with NetApp SnapMirror active sync with VMware vSphere Metro Storage Cluster (vMSC)

Use VMware vSphere 6.5 and 6.7 with ONTAP

You can use ONTAP SAN host configuration settings for the vSphere 6.5.x and 6.7.x releases with FC, FCoE and iSCSI protocols.

Hypervisor SAN Booting

Before you begin

If you decide to use SAN booting, it must be supported by your configuration. You can use the Interoperability Matrix Tool to verify that your OS, HBA, HBA firmware and the HBA boot BIOS, and ONTAP version are supported.

Steps

1. Map the SAN boot LUN to the host.
2. Verify that multiple paths are available.

   Multiple paths become available after the host OS is up and running on the paths.

3. Enable SAN booting in the server BIOS for the ports to which the SAN boot LUN is mapped.

   For information on how to enable the HBA BIOS, see your vendor-specific documentation.

4. Reboot the host to verify that the boot was successful.

Multipathing

ESXi provides an extensible multipathing module called Native Multipathing Plug-In (NMP) that manages the sub-plugins, Storage Array Type Plugins (SATPs), and Path Selection Plugins (PSPs). These SATP rules are available by default in ESXi.

For ONTAP storage, the "VMW_SATP_ALUA" plugin is used by default with "VMW_PSP_RR" as a path selection policy (PSP). You can run the following command to confirm the PSP:

`esxcli storage nmp satp rule list -s VMW_SATP_ALUA`

```
Name           Device  Vendor    Model     Driver  Transport  Options
-------------  ------  ------    -------   ------  ---------  -------
VMW_SATP_ALUA           LSI       INF-01-00
reset_on_attempted_reserve
VMW_SATP_ALUA           NETAPP
reset_on_attempted_reserve

Rule Group  Claim Options  Default PSP  PSP Options  Description
----------  -------------  -----------  -----------  ------------
system        tpgs_on         VMW_PSP_MRU       NetApp E-Series arrays
with ALUA support
system        tpgs_on          MW_PSP_RR          NetApp arrays with ALUA
support
```

Unresolved directive in hu_vsphere_6.adoc - include::_include/hu/reuse_hu_asa_configuration.adoc[]

esxcli storage nmp device list -d naa.600a098038304759563f4e7837574453

```
fc.20000024ff171d37:21000024ff171d37-fc.202300a098ea5e27:204a00a098ea5e27-
naa.600a098038304759563f4e7837574453
   Runtime Name: vmhba4:C0:T0:L9
   Device: naa.600a098038304759563f4e7837574453
   Device Display Name: NETAPP Fibre Channel Disk
(naa.600a098038304759563f4e7837574453)
   Group State: active
   Array Priority: 0
   Storage Array Type Path Config:
{TPG_id=1000,TPG_state=AO,RTP_id=6,RTP_health=UP}
   Path Selection Policy Path Config: PSP VMW_PSP_RR does not support path
configuration.

fc.20000024ff171d36:21000024ff171d36-fc.202300a098ea5e27:201d00a098ea5e27-
naa.600a098038304759563f4e7837574453
   Runtime Name: vmhba3:C0:T1:L9
   Device: naa.600a098038304759563f4e7837574453
   Device Display Name: NETAPP Fibre Channel Disk
(naa.600a098038304759563f4e7837574453)
   Group State: active
   Array Priority: 0
   Storage Array Type Path Config:
{TPG_id=1001,TPG_state=AO,RTP_id=3,RTP_health=UP}
   Path Selection Policy Path Config: PSP VMW_PSP_RR does not support path
configuration.

fc.20000024ff171d36:21000024ff171d36-fc.202300a098ea5e27:201b00a098ea5e27-
naa.600a098038304759563f4e7837574453
   Runtime Name: vmhba3:C0:T0:L9
   Device: naa.600a098038304759563f4e7837574453
   Device Display Name: NETAPP Fibre Channel Disk
(naa.600a098038304759563f4e7837574453)
   Group State: active
   Array Priority: 0
   Storage Array Type Path Config:
{TPG_id=1000,TPG_state=AO,RTP_id=1,RTP_health=UP}
   Path Selection Policy Path Config: PSP VMW_PSP_RR does not support path
configuration.

fc.20000024ff171d37:21000024ff171d37-fc.202300a098ea5e27:201e00a098ea5e27-
naa.600a098038304759563f4e7837574453
   Runtime Name: vmhba4:C0:T1:L9
```

```
   Device: naa.600a098038304759563f4e7837574453
   Device Display Name: NETAPP Fibre Channel Disk
(naa.600a098038304759563f4e7837574453)
   Group State: active
   Array Priority: 0
   Storage Array Type Path Config:
{TPG_id=1001,TPG_state=AO,RTP_id=4,RTP_health=UP}
   Path Selection Policy Path Config: PSP VMW_PSP_RR does not support path
configuration.
```

In the above example, LUN has been mapped from NetApp storage with 4 paths (4 active-optimized).

Non-ASA configurations

For non-ASA configurations, there should be two groups of paths with different priorities. The paths with higher priorities are Active/Optimized. This means they are serviced by the controller where the aggregate is located. The paths with lower priorities are active but non-optimized because they are served from a different controller. The non-optimized paths are only used when optimized paths are not available.

Example

The following example displays the correct output for an ONTAP LUN with two Active/Optimized paths and two Active/Non-Optimized paths.

`esxcli storage nmp path list -d naa.600a098038313530772b4d673979372f`

```
fc.20000090fae0ec8e:10000090fae0ec8e-fc.201000a098dfe3d1:200b00a098dfe3d1-
naa.600a098038313530772b4d673979372f
   Runtime Name: vmhba3:C0:T2:L21
   Device: naa.600a098038313530772b4d673979372f
   Device Display Name: NETAPP Fibre Channel Disk
(naa.600a098038313530772b4d673979372f)
   Group State: active unoptimized
   Array Priority: 0
   Storage Array Type Path Config:
{TPG_id=1001,TPG_state=ANO,RTP_id=29,RTP_health=UP}
   Path Selection Policy Path Config: PSP VMW_PSP_RR does not support path
configuration.

fc.20000090fae0ec8e:10000090fae0ec8e-fc.201000a098dfe3d1:200700a098dfe3d1-
naa.600a098038313530772b4d673979372f
   Runtime Name: vmhba3:C0:T3:L21
   Device: naa.600a098038313530772b4d673979372f
   Device Display Name: NETAPP Fibre Channel Disk
(naa.600a098038313530772b4d673979372f)
   Group State: active
   Array Priority: 0
   Storage Array Type Path Config:
```

```
{TPG_id=1000,TPG_state=AO,RTP_id=25,RTP_health=UP}
   Path Selection Policy Path Config: PSP VMW_PSP_RR does not support path
configuration.

fc.20000090fae0ec8f:10000090fae0ec8f-fc.201000a098dfe3d1:200800a098dfe3d1-
naa.600a098038313530772b4d673979372f
   Runtime Name: vmhba4:C0:T2:L21
   Device: naa.600a098038313530772b4d673979372f
   Device Display Name: NETAPP Fibre Channel Disk
(naa.600a098038313530772b4d673979372f)
   Group State: active
   Array Priority: 0
   Storage Array Type Path Config:
{TPG_id=1000,TPG_state=AO,RTP_id=26,RTP_health=UP}
   Path Selection Policy Path Config: PSP VMW_PSP_RR does not support path
configuration.

fc.20000090fae0ec8f:10000090fae0ec8f-fc.201000a098dfe3d1:200c00a098dfe3d1-
naa.600a098038313530772b4d673979372f
   Runtime Name: vmhba4:C0:T3:L21
   Device: naa.600a098038313530772b4d673979372f
   Device Display Name: NETAPP Fibre Channel Disk
(naa.600a098038313530772b4d673979372f)
   Group State: active unoptimized
   Array Priority: 0
   Storage Array Type Path Config:
{TPG_id=1001,TPG_state=ANO,RTP_id=30,RTP_health=UP}
   Path Selection Policy Path Config: PSP VMW_PSP_RR does not support path
configuration.
```

In the above example, LUN has been mapped from NetApp storage with 4 paths (2 active-optimized and 2 active-unoptimized).

vVol

Virtual Volumes (vVols) are a VMware object type that corresponds to a Virtual Machine (VM) disk, and its snapshots and fast clones.

ONTAP tools for VMware vSphere includes the VASA Provider for ONTAP, which provides the integration point for a VMware vCenter to leverage vVols based storage. When you deploy the ONTAP tools OVA, it is automatically registered with the vCenter server and enables the VASA Provider.

When you create a vVols datastore using the vCenter UI, it guides you to create FlexVols as backup storage for the datastore. vVols within a vVols datastores are accessed by ESXi hosts using a protocol endpoint (PE). In SAN environments, one 4MB LUN is created on each FlexVol volume volume volume in the datastore for use as a PE. A SAN PE is an administrative logical unit (ALU); vVols are subsidiary logical units (SLUs).

Standard requirements and best practices for SAN environments apply when using vVols, including (but not limited to) the following:

1. Create at least one SAN LIF on each node per SVM you intend to use. The best practice is to create at least two per node, but no more than necessary.
2. Eliminate any single point of failure. Use multiple VMkernel network interfaces on different network subnets that use NIC teaming when multiple virtual switches are used or use multiple physical NICs connected to multiple physical switches to provide HA and increased throughput.
3. Configure zoning and/or VLANs as required for host connectivity.
4. Ensure all required initiators are logged into the target LIFs on the desired SVM.

You must deploy ONTAP tools for VMware vSphere to enable the VASA Provider. The VASA Provider will manage all of your igroup settings for you, so there is no need to create or manage iGroups in a vVols environment.

NetApp doesn't recommend changing any vVols settings from default at this time.

Refer to the Interoperability Matrix Tool for specific versions of ONTAP tools, or legacy VASA Provider for your specific versions of vSphere and ONTAP.

For detailed information on provisioning and managing vVols, refer to the ONTAP tools for VMware vSphere documentation as well as VMware vSphere with ONTAP and Virtual Volumes (vVols) with ONTAP tools 10.

Recommended Settings

ATS Locking

ATS locking is **mandatory** for VAAI compatible storage and upgraded VMFS5 and is required for proper interoperability and optimal VMFS shared storage I/O performance with ONTAP LUNs. Refer to VMware documentation for details on enabling ATS locking.

Settings	Default	ONTAP Recommended	Description
HardwareAcceleratedLocking	1	1	Helps enable the use of Atomic Test and Set (ATS) locking
Disk IOPs	1000	1	IOPS limit: The Round Robin PSP defaults to an IOPS limit of 1000. In this default case, a new path is used after 1000 I/O operations are issued.
Disk/QFullSampleSize	0	32	The count of QUEUE FULL or BUSY conditions it takes before ESXi starts throttling.

Enable Space-alloc setting for all the LUN's mapped to VMware vSphere for UNMAP to work. For more details, refer to ONTAP Documentation.

Guest OS timeouts

You can manually configure the virtual machines with the recommended guest OS tunings. After tuning updates, you must reboot the guest for the updates to take effect.

GOS timeout values:

Guest OS Type	Timeouts
Linux variants	disk timeout = 60
Windows	disk timeout = 60
Solaris	disk timeout = 60 busy retry = 300 not ready retry = 300 reset retry = 30 max.throttle = 32 min.throttle = 8

Validate the vSphere tunable

Use the following command to verify the `HardwareAcceleratedLocking` setting:

`esxcli system settings advanced list --option /VMFS3/HardwareAcceleratedLocking`

```
   Path: /VMFS3/HardwareAcceleratedLocking
   Type: integer
   Int Value: 1
   Default Int Value: 1
   Min Value: 0
   Max Value: 1
   String Value:
   Default String Value:
   Valid Characters:
   Description: Enable hardware accelerated VMFS locking (requires
compliant hardware). Please see http://kb.vmware.com/kb/2094604 before
disabling this option.
```

Validate the disk IOPs setting

Use the following command to verify the IOPs setting:

`esxcli storage nmp device list -d naa.600a098038304731783f506670553355`

```
naa.600a0980383047317783f506670553355
   Device Display Name: NETAPP Fibre Channel Disk
(naa.600a0980383047317783f506670553355)
   Storage Array Type: VMW_SATP_ALUA
   Storage Array Type Device Config: {implicit_support=on;
explicit_support=off; explicit_allow=on; alua_followover=on;
action_OnRetryErrors=off;
{TPG_id=1000,TPG_state=ANO}{TPG_id=1001,TPG_state=AO}}
   Path Selection Policy: VMW_PSP_RR
   Path Selection Policy Device Config:
{policy=rr,iops=1,bytes=10485760,useANO=0; lastPathIndex=0:
NumIOsPending=0,numBytesPending=0}
   Path Selection Policy Device Custom Config:
   Working Paths: vmhba4:C0:T0:L82, vmhba3:C0:T0:L82
   Is USB: false
```

Validate the QFullSampleSize

Use the following command to verify the QFullSampleSize:

`esxcli system settings advanced list --option /Disk/QFullSampleSize`

```
   Path: /Disk/QFullSampleSize
   Type: integer
   Int Value: 32
   Default Int Value: 0
   Min Value: 0
   Max Value: 64
   String Value:
   Default String Value:
   Valid Characters:
   Description: Default I/O samples to monitor for detecting non-transient
queue full condition. Should be nonzero to enable queue depth throttling.
Device specific QFull options will take precedence over this value if set.
```

Known issues

The VMware vSphere 6.5 and 6.7 with ONTAP release has the following known issues:

OS version	NetApp Bug ID	Title	Description
ESXi 6.5 and ESXi 6.7.x	1413424	WFC RDM luns fails during testing	Windows failover clustering raw device mapping between Windows Virtual Machines like Windows 2019, Windows 2016, and Windows 2012 across VMWare ESXi host failed during storage failover testing on all the 7-mode, C-cmode cluster controllers.
ESXi 6.5.x and ESXi 6.7.x	1256473	PLOGI issue seen during testing on Emulex adapters	

Related information

- VMware vSphere with ONTAP
- VMware vSphere 5.x, 6.x and 7.x support with NetApp MetroCluster (2031038)
- NetApp ONTAP with NetApp SnapMirror active sync with VMware vSphere Metro Storage Cluster (vMSC)

HP-UX

Configure HP-UX 11i v3 for FCP and iSCSI with ONTAP storage

The HP-UX Host Utilities software provides management and diagnostic tools for HP-UX hosts that are connected to ONTAP storage. When you install the HP-UX Host Utilities on a HP-UX 11i v3 host, you can use the Host Utilities to help you manage FCP and iSCSI protocol operations with ONTAP LUNs.

Step 1: Optionally, enable SAN booting

Configure your host to use SAN booting to simplify deployment and improve scalability. SAN booting is the process of setting up a SAN-attached disk (a LUN) as a boot device for a HP-UX host. The Host Utilities support SAN booting with FC and FCoE protocols in HP-UX environments.

Steps

1. Use the Interoperability Matrix Tool to verify that your Linux OS, host bus adapter (HBA), HBA firmware, HBA boot BIOS, and ONTAP version support SAN booting.
2. Follow the best practices for setting up a SAN boot in the HP-UX vendor documentation.

Step 2: Install the HP-UX Host Utilities

NetApp strongly recommends installing the HP-UX Host Utilities to support ONTAP LUN management and assist technical support with gathering configuration data.

Install HP-UX Host Utilities 6.0

Step 3: Confirm the multipath configuration for your host

Use multipathing with HP-UX 11i v3 to manage ONTAP LUNs. Multipathing allows you to configure multiple network paths between the host and storage system. If one path fails, traffic continues on the remaining paths.

After installing HP-UX Host Utilities, verify that you have the NetApp recommended settings configured for your ONTAP LUNs.

About this task

The HP-UX Host Utilities support Native Microsoft Multipath I/O (MPIO) and Veritas Dynamic Multipathing. The following steps are for the Native MPIO solution.

Steps

1. When you install the HP-UX Host Utilities, the following recommended default settings are automatically loaded for ONTAP LUNs.

   Show parameter settings

Parameter	Uses Default Value
transient_secs	120
leg_mpath_enable	TRUE
max_q_depth	8
path_fail_secs	120
load_bal_policy	Round_robin
lua_enabled	TRUE
esd_secs	30

2. Verify the parameter settings and path status for your ONTAP LUNs:

   ```
   sanlun lun show
   ```

   The default multipath parameters support ASA, AFF, and FAS configurations. The following example outputs show the correct parameter settings and path status for ONTAP LUNs in an ASA, AFF, or FAS configuration.

ASA configuration

An ASA configuration optimizes all paths to a given LUN, keeping them active. This improves performance by serving I/O operations through all paths at the same time.

Show example

```
# sanlun lun show -p vs39:/vol/hpux_vol_1_1/hpux_lun

                    ONTAP Path: vs39:/vol/hpux_vol_1_1/hpux_lun
                           LUN: 2
                      LUN Size: 30g
                   Host Device: /dev/rdisk/disk25
                          Mode: C
              Multipath Provider: None
-------- ---------- -------------------- ------- --------------
host     vserver    /dev/dsk
path     path       filename             host    vserver
state    type       or hardware path     adapter LIF
-------- ---------- -------------------- ------- --------------
up       primary    /dev/dsk/c4t0d2      fcd0    248_1c_hp
up       primary    /dev/dsk/c6t0d2      fcd0    246_1c_hp
up       primary    /dev/dsk/c10t0d2     fcd1    246_1d_hp
up       primary    /dev/dsk/c8t0d2      fcd1    248_1d_hp
```

AFF or FAS configuration

An AFF or FAS configuration should have two groups of paths with higher and lower priorities. Higher priority Active/Optimized paths are served by the controller where the aggregate is located. Lower priority paths are active but non-optimized because they are served by a different controller. Non-optimized paths are only used when optimized paths aren’t available.

The following example displays the output for an ONTAP LUN with two Active/Optimized paths and two Active/Non-Optimized paths:

Show example

```
# sanlun lun show -p vs39:/vol/vol24_3_0/lun24_0
                    ONTAP Path: vs39:/vol/vol24_3_0/lun24_0
                           LUN: 37
                      LUN Size: 15g
                   Host Device: /dev/rdisk/disk942
                          Mode: C
              Multipath Policy: A/A
            Multipath Provider: Native
------- ---------- ------ ------- ---------- ----------
--------------
host    vserver     /dev/dsk                             HP A/A
path    path        filename          host    vserver   path
failover
state   type        or hardware path adapter    LIF     priority
------- ---------- ------ ------- ---------- ----------
--------------
up       primary    /dev/dsk/c39t4d5   fcd0    hpux_3     0
up       primary    /dev/dsk/c41t4d5   fcd1    hpux_4     0
up       secondary  /dev/dsk/c40t4d5   fcd0    hpux_3     1
up       secondary  /dev/dsk/c42t4d5   fcd1    hpux_4     1
```

Step 4: Review the known issues

The HP-UX 11i v3 with ONTAP storage release has the following known issues:

NetApp Bug ID	Title	Description	Partner ID
1447287	AUFO event on the isolated master cluster in SnapMirror active sync configuration causes temporary disruption on the HP-UX host	This issue occurs when there is an automatic unplanned failover (AUFO) event on the isolated master cluster in the SnapMirror active sync configuration. It might take more than 120 seconds for I/O to resume on the HP-UX host, but this might not cause any I/O disruption or error messages. This issue causes dual event failure because the connection between the primary and the secondary cluster is lost and the connection between the primary cluster and the mediator is also lost. This is considered a rare event, unlike other AUFO events.	NA
1344935	HP-UX 11.31 Host intermittently reporting path status incorrectly on ASA setup.	Path reporting issues with ASA configuration.	NA
1306354	HP-UX LVM creation sends I/O of block size above 1MB	SCSI Maximum Transfer Length of 1MB is enforced in ONTAP All SAN Array. To restrict the Maximum Transfer Length from HP-UX hosts when connected to ONTAP All SAN Array, it is required to set the Maximum I/O size allowed by the HP-UX SCSI subsystem to 1MB. Refer HP-UX vendor documentation for details.	NA

What's next

Learn about using the HP-UX Host Utilities tool.

HPE VME

Configure HPE VME 8.0.x for FCP and iSCSI with ONTAP storage

The Linux Host Utilities software provides management and diagnostic tools for Linux hosts that are connected to ONTAP storage. When you install the Linux Host Utilities on a HPE VME 8.0.x host, you can use the Host Utilities to help you manage FCP and iSCSI protocol operations with ONTAP LUNs.

Step 1: Install the Linux Host Utilities

NetApp strongly recommends installing the Linux Host Utilities to support ONTAP LUN management and assist technical support with gathering configuration data.

Install Linux Host Utilities 8.0.

Installing the Linux Host Utilities doesn't change any host timeout settings on your Linux host.

Step 2: Confirm the multipath configuration for your host

You can use multipathing with HPE VME 8.0.x to manage ONTAP LUNs.

To ensure that multipathing is configured correctly for your host, verify that the `/etc/multipath.conf` file is defined and that you have the NetApp recommended settings configured for your ONTAP LUNs.

Steps

1. Verify that the `/etc/multipath.conf` file exits. If the file doesn't exist, create an empty, zero-byte file:

```
touch /etc/multipath.conf
```

2. The first time the `multipath.conf` file is created, you might need to enable and start the multipath services to load the recommended settings:

```
systemctl enable multipathd
```

```
systemctl start multipathd
```

3. Each time you boot the host, the empty `/etc/multipath.conf` zero-byte file automatically loads the NetApp recommended host multipath parameters as the default settings. You shouldn't need to make changes to the `/etc/multipath.conf` file for your host because the operating system is compiled with the multipath parameters that recognize and manage ONTAP LUNs correctly.

   The following table shows the Linux OS native compiled multipath parameter settings for ONTAP LUNs.

Show parameter settings

Parameter	Setting
detect_prio	yes
dev_loss_tmo	"infinity"
failback	immediate
fast_io_fail_tmo	5
features	"2 pg_init_retries 50"
flush_on_last_del	"yes"
hardware_handler	"0"
no_path_retry	queue
path_checker	"tur"
path_grouping_policy	"group_by_prio"
path_selector	"service-time 0"
polling_interval	5
prio	"ontap"
product	LUN
retain_attached_hw_handler	yes
rr_weight	"uniform"
user_friendly_names	no
vendor	NETAPP

4. Verify the parameter settings and path status for your ONTAP LUNs:

```
multipath -ll
```

The default multipath parameters support ASA, AFF, and FAS configurations. In these configurations, a single ONTAP LUN shouldn’t require more than four paths. Having more than four paths can cause problems during a storage failure.

The following example outputs show the correct parameter settings and path status for ONTAP LUNs in an ASA, AFF, or FAS configuration.

ASA configuration

An ASA configuration optimizes all paths to a given LUN, keeping them active. This improves performance by serving I/O operations through all paths at the same time.

Show example

```
# multipath -ll
3600a098038314c4a433f577471797958 dm-2 NETAPP,LUN C-Mode
size=180G features='3 queue_if_no_path pg_init_retries 50'
hwhandler='1 alua' wp=rw
`-+- policy='service-time 0' prio=50 status=active
  |- 14:0:0:0  sdc  8:32   active ready running
  |- 17:0:0:0  sdas 66:192 active ready running
  |- 14:0:3:0  sdar 66:176 active ready running
  `- 17:0:3:0  sdch 69:80  active ready running
```

AFF or FAS configuration

An AFF or FAS configuration should have two groups of paths with higher and lower priorities. Higher priority Active/Optimized paths are served by the controller where the aggregate is located. Lower priority paths are active but non-optimized because they are served by a different controller. Non-optimized paths are only used when optimized paths aren't available.

The following example displays the output for an ONTAP LUN with two Active/Optimized paths and two Active/Non-Optimized paths:

Show example

```
# multipath -ll
3600a0980383149764b5d567257516273 dm-0 NETAPP,LUN C-Mode
size=150G features='3 queue_if_no_path pg_init_retries 50'
hwhandler='1 alua' wp=rw
|-+- policy='service-time 0' prio=50 status=active
| |- 16:0:3:0  sdcg 69:64  active ready running
| `- 10:0:0:0  sdb  8:16   active ready running
`-+- policy='service-time 0' prio=10 status=enabled
  |- 10:0:1:0  sdc  8:32   active ready running
  `- 16:0:2:0  sdcf 69:48  active ready running
```

Step 3: Confirm the iSCSI configuration for your host

Ensure that iSCSI is configured correctly for your host.

About this task

You perform the following steps on the iSCSI host.

Steps

1. Verify that the iSCSI initiator package (open-iscsi) is installed:

   ```
   $apt list |grep open-iscsi
   ```

   You should see an output similar to the following example:

   ```
   open-iscsi/noble-updates,noble-updates,now 2.1.9-3ubuntu5.4 amd64
   ```

2. Verify the iSCSI initiator node name, which is located in the `/etc/iscsi/initiatorname.iscsi` file:

   ```
   InitiatorName=iqn.YYYY-MM.com.<vendor>:<host_name>
   ```

3. Configure the iSCSI session timeout parameter located in the `/etc/iscsi/iscsid.conf` file:

   ```
   node.session.timeo.replacement_timeout = 5
   ```

   The iSCSI `replacement_timeout` parameter controls how long the iSCSI layer should wait for a timed-out path or session to reestablish itself before failing any commands on it. You should set the value of `replacement_timeout` to 5 in the iSCSI configuration file.

4. Enable the iSCSI service:

   ```
   $systemctl enable iscsid
   ```

5. Start the iSCSI service:

   ```
   $systemctl start iscsid
   ```

6. Verify that the iSCSI service is running:

   ```
   $systemctl status iscsid
   ```

Show example

```
●iscsid.service - iSCSI initiator daemon (iscsid)
     Loaded: loaded (/usr/lib/systemd/system/iscsid.service;
enabled; preset: disabled)
     Active: active (running) since Mon 2026-01-12 12:53:18 IST; 2
days ago
TriggeredBy: ● iscsid.socket
       Docs: man:iscsid(8)
   Main PID: 1127419 (iscsid)
      Tasks: 2 (limit: 76557)
     Memory: 4.3M (peak: 8.8M)
        CPU: 1.657s
     CGroup: /system.slice/iscsid.service
             ├─1127418 /usr/sbin/iscsid
             └─1127419 /usr/sbin/iscsid
```

7. Discover the iSCSI targets:

```
$iscsiadm --mode discovery --op update --type sendtargets --portal
<target_IP>
```

show example

```
iscsiadm --mode discovery --op update --type sendtargets --portal
192.168.100.197
192.168.100.197:3260,1046 iqn.1992-
08.com.netapp:sn.7cd154a7d35411f0a25ed039eaa95f59:vs.8
192.168.200.199:3260,1049 iqn.1992-
08.com.netapp:sn.7cd154a7d35411f0a25ed039eaa95f59:vs.8
192.168.100.199:3260,1048 iqn.1992-
08.com.netapp:sn.7cd154a7d35411f0a25ed039eaa95f59:vs.8
192.168.200.197:3260,1047 iqn.1992-
08.com.netapp:sn.7cd154a7d35411f0a25ed039eaa95f59:vs.8
```

8. Log in to the targets:

```
$iscsiadm --mode node -l all
```

9. Set iSCSI to log in automatically when the host boots:

```
$iscsiadm --mode node -T <target_name> -p <ip:port> -o update -n
node.startup -v automatic
```

You should see an output similar to the following example:

```
iscsiadm --mode node -T iqn.1992-
08.com.netapp:sn.7cd154a7d35411f0a25ed039eaa95f59:vs.8 -p
192.168.100.197:3260 -o update -n node.startup -v automatic
```

10. Verify the iSCSI sessions:

```
$iscsiadm --mode session
```

Show example

```
iscsiadm --mode session
tcp: [1] 192.168.200.197:3260,1047 iqn.1992-
08.com.netapp:sn.7cd154a7d35411f0a25ed039eaa95f59:vs.8 (non-flash)
tcp: [2] 192.168.100.197:3260,1046 iqn.1992-
08.com.netapp:sn.7cd154a7d35411f0a25ed039eaa95f59:vs.8 (non-flash)
tcp: [3] 192.168.100.199:3260,1048 iqn.1992-
08.com.netapp:sn.7cd154a7d35411f0a25ed039eaa95f59:vs.8 (non-flash)
tcp: [4] 192.168.200.199:3260,1049 iqn.1992-
08.com.netapp:sn.7cd154a7d35411f0a25ed039eaa95f59:vs.8 (non-flash)
```

Step 4: Optionally, exclude a device from multipathing

If required, you can exclude a device from multipathing by adding the WWID for the unwanted device to the "blacklist" stanza for the `multipath.conf` file.

Steps

1. Determine the WWID:

```
/lib/udev/scsi_id -gud /dev/sda
```

"sda" is the local SCSI disk that you want to add to the blacklist.

An example WWID is `360030057024d0730239134810c0cb833`.

2. Add the WWID to the "blacklist" stanza:

```
blacklist {
         wwid   360030057024d0730239134810c0cb833
        devnode "^(ram|raw|loop|fd|md|dm-|sr|scd|st)[0-9]*"
        devnode "^hd[a-z]"
        devnode "^cciss.*"
}
```

Step 5: Customize multipath parameters for ONTAP LUNs

If your host is connected to LUNs from other vendors and any of the multipath parameter settings are overridden, you need to correct them by adding stanzas later in the `multipath.conf` file that apply specifically to ONTAP LUNs. If you don't do this, the ONTAP LUNs might not work as expected.

Check your `/etc/multipath.conf` file, especially in the defaults section, for settings that might be overriding the default settings for multipath parameters.

> ⚠ You shouldn't override the recommended parameter settings for ONTAP LUNs. These settings are required for optimal performance of your host configuration. Contact NetApp support, your OS vendor, or both for more information.

The following example shows how to correct an overridden default. In this example, the `multipath.conf` file defines values for `path_checker` and `no_path_retry` that aren't compatible with ONTAP LUNs, and you can't remove these parameters because ONTAP storage arrays are still attached to the host. Instead, you correct the values for `path_checker` and `no_path_retry` by adding a device stanza to the `multipath.conf` file that applies specifically to the ONTAP LUNs.

Show example

```
defaults {
   path_checker       readsector0
   no_path_retry      fail
}

devices {
   device {
      vendor           "NETAPP"
      product          "LUN"
      no_path_retry    queue
      path_checker     tur
   }
}
```

Step 6: Review the known issues

There are no known issues.

What's next?

- Learn about using the Linux Host Utilities tool.
- Configure HPE VME data stores on the ONTAP LUNs, then configure the virtual machine. Refer to the vendor HPE documentation for more information.

Oracle Linux

Configure Oracle Linux 9.x for FCP and iSCSI with ONTAP storage

The Linux Host Utilities software provides management and diagnostic tools for Linux hosts that are connected to ONTAP storage. When you install the Linux Host Utilities on an Oracle Linux 9.x host, you can use the Host Utilities to help you manage FCP and iSCSI protocol operations with ONTAP LUNs.

Step 1: Optionally, enable SAN booting

You can configure your host to use SAN booting to simplify deployment and improve scalability.

Before you begin

Use the Interoperability Matrix Tool to verify that your Linux OS, host bus adapter (HBA), HBA firmware, HBA boot BIOS, and ONTAP version support SAN booting.

Steps

1. Create a SAN boot LUN and map it to the host.
2. Enable SAN booting in the server BIOS for the ports to which the SAN boot LUN is mapped.

   For information on how to enable the HBA BIOS, see your vendor-specific documentation.

3. Verify that the configuration was successful by rebooting the host and verifying that the OS is up and running.

Step 2: Install the Linux Host Utilities

NetApp strongly recommends installing the Linux Host Utilities to support ONTAP LUN management and assist technical support with gathering configuration data.

Install Linux Host Utilities 8.0.

Installing the Linux Host Utilities doesn't change any host timeout settings on your Linux host.

Step 3: Confirm the multipath configuration for your host

You can use multipathing with Oracle Linux 9.x to manage ONTAP LUNs.

To ensure that multipathing is configured correctly for your host, verify that the `/etc/multipath.conf` file is defined and that you have the NetApp recommended settings configured for your ONTAP LUNs.

Steps

1. Verify that the `/etc/multipath.conf` file exits. If the file doesn't exist, create an empty, zero-byte file:

```
touch /etc/multipath.conf
```

2. The first time the `multipath.conf` file is created, you might need to enable and start the multipath services to load the recommended settings:

```
systemctl enable multipathd
```

```
systemctl start multipathd
```

3. Each time you boot the host, the empty `/etc/multipath.conf` zero-byte file automatically loads the NetApp recommended host multipath parameters as the default settings. You shouldn't need to make changes to the `/etc/multipath.conf` file for your host because the operating system is compiled with the multipath parameters that recognize and manage ONTAP LUNs correctly.

   The following table shows the Linux OS native compiled multipath parameter settings for ONTAP LUNs.

   Show parameter settings

Parameter	Setting
detect_prio	yes
dev_loss_tmo	"infinity"
failback	immediate
fast_io_fail_tmo	5
features	"2 pg_init_retries 50"
flush_on_last_del	"yes"
hardware_handler	"0"
no_path_retry	queue
path_checker	"tur"
path_grouping_policy	"group_by_prio"
path_selector	"service-time 0"
polling_interval	5
prio	"ontap"
product	LUN
retain_attached_hw_handler	yes
rr_weight	"uniform"
user_friendly_names	no
vendor	NETAPP

4. Verify the parameter settings and path status for your ONTAP LUNs:

```
multipath -ll
```

The default multipath parameters support ASA, AFF, and FAS configurations. In these configurations, a single ONTAP LUN shouldn’t require more than four paths. Having more than four paths can cause problems during a storage failure.

The following example outputs show the correct parameter settings and path status for ONTAP LUNs in an ASA, AFF, or FAS configuration.

ASA configuration

An ASA configuration optimizes all paths to a given LUN, keeping them active. This improves performance by serving I/O operations through all paths at the same time.

Show example

```
multipath -ll
3600a098038303634722b4d59646c4436 dm-28 NETAPP,LUN C-Mode
size=10G features='3 queue_if_no_path pg_init_retries 50'
hwhandler='1 alua' wp=rw
|-+- policy='service-time 0' prio=50 status=active
  |- 11:0:7:6   sdbz 68:208  active ready running
  |- 11:0:11:6  sddn 71:80   active ready running
  |- 11:0:15:6  sdfb 129:208 active ready running
  |- 12:0:1:6   sdgp 132:80  active ready running
```

AFF or FAS configuration

An AFF or FAS configuration should have two groups of paths with higher and lower priorities. Higher priority Active/Optimized paths are served by the controller where the aggregate is located. Lower priority paths are active but non-optimized because they are served by a different controller. Non-optimized paths are only used when optimized paths aren’t available.

The following example displays the output for an ONTAP LUN with two Active/Optimized paths and two Active/Non-Optimized paths:

Show example

```
multipath -ll
3600a0980383036347ffb4d59646c4436 dm-28 NETAPP,LUN C-Mode
size=10G features='3 queue_if_no_path pg_init_retries 50'
hwhandler='1 alua' wp=rw
|-+- policy='service-time 0' prio=50 status=active
| |- 16:0:6:35 sdwb  69:624  active ready running
| |- 16:0:5:35 sdun  66:752  active ready running
`-+- policy='service-time 0' prio=10 status=enabled
  |- 15:0:0:35 sdaj  66:48   active ready running
  |- 15:0:1:35 sdbx  68:176  active ready running
```

Step 4: Confirm the iSCSI configuration for your host

Ensure that iSCSI is configured correctly for your host.

About this task

You perform the following steps on the iSCSI host.

Steps

1. Verify that the iSCSI initiator package (iscsi-initiator-utils) is installed:

   ```
   rpm -qa | grep iscsi-initiator-utils
   ```

   You should see an output similar to the following example:

   ```
   iscsi-initiator-utils-6.2.1.11-0.git4b3e853.el9.x86_64
   ```

2. Verify the iSCSI initiator node name, which is located in the `/etc/iscsi/initiatorname.iscsi` file:

   ```
   InitiatorName=iqn.YYYY-MM.com.<vendor>:<host_name>
   ```

3. Configure the iSCSI session timeout parameter located in the `/etc/iscsi/iscsid.conf` file:

   ```
   node.session.timeo.replacement_timeout = 5
   ```

   The iSCSI `replacement_timeout` parameter controls how long the iSCSI layer should wait for a timed-out path or session to reestablish itself before failing any commands on it. You should set the value of `replacement_timeout` to 5 in the iSCSI configuration file.

4. Enable the iSCSI service:

   ```
   $systemctl enable iscsid
   ```

5. Start the iSCSI service:

   ```
   $systemctl start iscsid
   ```

6. Verify that the iSCSI service is running:

   ```
   $systemctl status iscsid
   ```

Show example

```
● iscsid.service - Open-iSCSI
     Loaded: loaded (/usr/lib/systemd/system/iscsid.service;
enabled; preset: disabled)
     Active: active (running) since Tue 2025-12-02 11:36:21 EST; 2
weeks 1 day ago
TriggeredBy: ● iscsid.socket
       Docs: man:iscsid(8)
             man:iscsiuio(8)
             man:iscsiadm(8)
   Main PID: 2263 (iscsid)
     Status: "Ready to process requests"
      Tasks: 1 (limit: 816061)
     Memory: 18.5M
        CPU: 14.480s
     CGroup: /system.slice/iscsid.service
             └─2263 /usr/sbin/iscsid -f -d2
```

7. Discover the iSCSI targets:

```
$iscsiadm --mode discovery --op update --type sendtargets --portal
<target_IP>
```

show example

```
iscsiadm --mode discovery --op update --type sendtargets --portal
192.168.30.87
192.168.30.87:3260,1139 iqn.1992-
08.com.netapp:sn.064a9b19b3ee11f09dcad039eabac370:vs.23
192.168.31.97:3260,1142 iqn.1992-
08.com.netapp:sn.064a9b19b3ee11f09dcad039eabac370:vs.23
192.168.31.87:3260,1141 iqn.1992-
08.com.netapp:sn.064a9b19b3ee11f09dcad039eabac370:vs.23
192.168.30.97:3260,1140 iqn.1992-
08.com.netapp:sn.064a9b19b3ee11f09dcad039eabac370:vs.23
```

8. Log in to the targets:

```
$iscsiadm --mode node -l all
```

9. Set iSCSI to log in automatically when the host boots:

```
$iscsiadm --mode node -T <target_name> -p <ip:port> -o update -n
node.startup -v automatic
```

You should see an output similar to the following example:

```
iscsiadm --mode node -T iqn.1992-
08.com.netapp:sn.064a9b19b3ee11f09dcad039eabac370:vs.23 -p
192.168.30.87:3260 -o update -n node.startup -v automatic
```

10. Verify the iSCSI sessions:

```
$iscsiadm --mode session
```

Show example

```
iscsiadm --mode session
tcp: [1] 192.168.30.87:3260,1139 iqn.1992-
08.com.netapp:sn.064a9b19b3ee11f09dcad039eabac370:vs.23 (non-flash)
tcp: [2] 192.168.31.97:3260,1142 iqn.1992-
08.com.netapp:sn.064a9b19b3ee11f09dcad039eabac370:vs.23 (non-flash)
tcp: [3] 192.168.31.87:3260,1141 iqn.1992-
08.com.netapp:sn.064a9b19b3ee11f09dcad039eabac370:vs.23 (non-flash)
tcp: [4] 192.168.30.97:3260,1140 iqn.1992-
08.com.netapp:sn.064a9b19b3ee11f09dcad039eabac370:vs.23 (non-flash)
```

Step 5: Optionally, exclude a device from multipathing

If required, you can exclude a device from multipathing by adding the WWID for the unwanted device to the "blacklist" stanza for the `multipath.conf` file.

Steps

1. Determine the WWID:

```
/lib/udev/scsi_id -gud /dev/sda
```

"sda" is the local SCSI disk that you want to add to the blacklist.

An example WWID is `360030057024d0730239134810c0cb833`.

2. Add the WWID to the "blacklist" stanza:

```
blacklist {
         wwid   360030057024d0730239134810c0cb833
        devnode "^(ram|raw|loop|fd|md|dm-|sr|scd|st)[0-9]*"
        devnode "^hd[a-z]"
        devnode "^cciss.*"
}
```

Step 6: Customize multipath parameters for ONTAP LUNs

If your host is connected to LUNs from other vendors and any of the multipath parameter settings are overridden, you need to correct them by adding stanzas later in the `multipath.conf` file that apply specifically to ONTAP LUNs. If you don't do this, the ONTAP LUNs might not work as expected.

Check your `/etc/multipath.conf` file, especially in the defaults section, for settings that might be overriding the default settings for multipath parameters.

> ⚠ You shouldn't override the recommended parameter settings for ONTAP LUNs. These settings are required for optimal performance of your host configuration. Contact NetApp support, your OS vendor, or both for more information.

The following example shows how to correct an overridden default. In this example, the `multipath.conf` file defines values for `path_checker` and `no_path_retry` that aren't compatible with ONTAP LUNs, and you can't remove these parameters because ONTAP storage arrays are still attached to the host. Instead, you correct the values for `path_checker` and `no_path_retry` by adding a device stanza to the `multipath.conf` file that applies specifically to the ONTAP LUNs.

Show example

```
defaults {
   path_checker       readsector0
   no_path_retry      fail
}

devices {
   device {
      vendor           "NETAPP"
      product          "LUN"
      no_path_retry    queue
      path_checker     tur
   }
}
```

Step 7: Review the known issues

Oracle Linux 9.3, 9.2, 9.1, and 9.0 with ONTAP storage has the following known issue:

NetApp Bug ID	Title	Description
1508554	SAN LUN utility with Emulex HBA needs symbolic links from library packages	When you execute the Linux Host Utilities CLI command - "sanlun fcp show adapter -v" on a SAN host, the command fails with an error message displaying that the library dependencies required for an host bus adapter (HBA) discovery cannot be located: ---- [root@hostname ~]# sanlun fcp show adapter -v Unable to locate /usr/lib64/libHBAAPI.so library Make sure the package installing the library is installed & loaded ----

What's next?

- Learn about using the Linux Host Utilities tool.
- Learn about ASM mirroring

  Automatic Storage Management (ASM) mirroring might require changes to the Linux multipath settings to allow ASM to recognize a problem and switch over to an alternate failure group. Most ASM configurations on ONTAP use external redundancy, which means that data protection is provided by the external array and ASM doesn't mirror data. Some sites use ASM with normal redundancy to provide two-way mirroring, normally across different sites. See Oracle databases on ONTAP for further information.

- Learn about Oracle Linux KVM and Virtualization

  Oracle Linux can serve as a KVM host. This enables you to run multiple virtual machines on a single physical server using the Linux Kernel-based Virtual Machine (KVM) technology. You can manage and support multiple Oracle Linux KVM hosts by using Oracle Linux Virtualization Manager, which is built from the open-source oVirt project. The KVM host doesn't require explicit host configuration settings for ONTAP LUNs.

Configure Oracle Linux 8.x for FCP and iSCSI with ONTAP storage

The Linux Host Utilities software provides management and diagnostic tools for Linux hosts that are connected to ONTAP storage. When you install the Linux Host Utilities on an Oracle Linux 8.x host, you can use the Host Utilities to help you manage FCP and iSCSI protocol operations with ONTAP LUNs.

Step 1: Optionally, enable SAN booting

You can configure your host to use SAN booting to simplify deployment and improve scalability.

Before you begin

Use the Interoperability Matrix Tool to verify that your Linux OS, host bus adapter (HBA), HBA firmware, HBA boot BIOS, and ONTAP version support SAN booting.

Steps

1. Create a SAN boot LUN and map it to the host.
2. Enable SAN booting in the server BIOS for the ports to which the SAN boot LUN is mapped.

   For information on how to enable the HBA BIOS, see your vendor-specific documentation.

3. Verify that the configuration was successful by rebooting the host and verifying that the OS is up and running.

Step 2: Install the Linux Host Utilities

NetApp strongly recommends installing the Linux Host Utilities to support ONTAP LUN management and assist technical support with gathering configuration data.

Install Linux Host Utilities 8.0.

Installing the Linux Host Utilities doesn't change any host timeout settings on your Linux host.

Step 3: Confirm the multipath configuration for your host

You can use multipathing with Oracle Linux 8.x to manage ONTAP LUNs.

You can use the recommended settings for Red Hat Enterprise Linux (RHEL) 8.x to configure Red Hat Compatible Kernel for Oracle Linux 8.x.

To ensure that multipathing is configured correctly for your host, verify that the `/etc/multipath.conf` file is defined and that you have the NetApp recommended settings configured for your ONTAP LUNs.

Steps

1. Verify that the `/etc/multipath.conf` file exits. If the file doesn't exist, create an empty, zero-byte file:

   ```
   touch /etc/multipath.conf
   ```

2. The first time the `multipath.conf` file is created, you might need to enable and start the multipath services to load the recommended settings:

   ```
   systemctl enable multipathd
   ```

   ```
   systemctl start multipathd
   ```

3. Each time you boot the host, the empty `/etc/multipath.conf` zero-byte file automatically loads the NetApp recommended host multipath parameters as the default settings. You shouldn't need to make changes to the `/etc/multipath.conf` file for your host because the operating system is compiled with the multipath parameters that recognize and manage ONTAP LUNs correctly.

The following table shows the Linux OS native compiled multipath parameter settings for ONTAP LUNs.

Show parameter settings

Parameter	Setting
detect_prio	yes
dev_loss_tmo	"infinity"
failback	immediate
fast_io_fail_tmo	5
features	"2 pg_init_retries 50"
flush_on_last_del	"yes"
hardware_handler	"0"
no_path_retry	queue
path_checker	"tur"
path_grouping_policy	"group_by_prio"
path_selector	"service-time 0"
polling_interval	5
prio	"ontap"
product	LUN
retain_attached_hw_handler	yes
rr_weight	"uniform"
user_friendly_names	no
vendor	NETAPP

4. Verify the parameter settings and path status for your ONTAP LUNs:

```
multipath -ll
```

The default multipath parameters support ASA, AFF, and FAS configurations. In these configurations, a single ONTAP LUN shouldn’t require more than four paths. Having more than four paths can cause problems during a storage failure.

The following example outputs show the correct parameter settings and path status for ONTAP LUNs in an ASA, AFF, or FAS configuration.

ASA configuration

An ASA configuration optimizes all paths to a given LUN, keeping them active. This improves performance by serving I/O operations through all paths at the same time.

Show example

```
multipath -ll
3600a098038303634722b4d59646c4436 dm-28 NETAPP,LUN C-Mode
size=10G features='3 queue_if_no_path pg_init_retries 50'
hwhandler='1 alua' wp=rw
|-+- policy='service-time 0' prio=50 status=active
  |- 11:0:7:6   sdbz 68:208  active ready running
  |- 11:0:11:6  sddn 71:80   active ready running
  |- 11:0:15:6  sdfb 129:208 active ready running
  |- 12:0:1:6   sdgp 132:80  active ready running
```

AFF or FAS configuration

An AFF or FAS configuration should have two groups of paths with higher and lower priorities. Higher priority Active/Optimized paths are served by the controller where the aggregate is located. Lower priority paths are active but non-optimized because they are served by a different controller. Non-optimized paths are only used when optimized paths aren’t available.

The following example displays the output for an ONTAP LUN with two Active/Optimized paths and two Active/Non-Optimized paths:

Show example

```
multipath -ll
3600a0980383036347ffb4d59646c4436 dm-28 NETAPP,LUN C-Mode
size=10G features='3 queue_if_no_path pg_init_retries 50'
hwhandler='1 alua' wp=rw
|-+- policy='service-time 0' prio=50 status=active
| |- 16:0:6:35 sdwb  69:624  active ready running
| |- 16:0:5:35 sdun  66:752  active ready running
`-+- policy='service-time 0' prio=10 status=enabled
  |- 15:0:0:35 sdaj  66:48   active ready running
  |- 15:0:1:35 sdbx  68:176  active ready running
```

Step 4: Confirm the iSCSI configuration for your host

Ensure that iSCSI is configured correctly for your host.

About this task

You perform the following steps on the iSCSI host.

Steps

1. Verify that the iSCSI initiator package (iscsi-initiator-utils) is installed:

   ```
   rpm -qa | grep iscsi-initiator-utils
   ```

   You should see an output similar to the following example:

   ```
   iscsi-initiator-utils-6.2.1.11-0.git4b3e853.el9.x86_64
   ```

2. Verify the iSCSI initiator node name, which is located in the `/etc/iscsi/initiatorname.iscsi` file:

   ```
   InitiatorName=iqn.YYYY-MM.com.<vendor>:<host_name>
   ```

3. Configure the iSCSI session timeout parameter located in the `/etc/iscsi/iscsid.conf` file:

   ```
   node.session.timeo.replacement_timeout = 5
   ```

   The iSCSI `replacement_timeout` parameter controls how long the iSCSI layer should wait for a timed-out path or session to reestablish itself before failing any commands on it. You should set the value of `replacement_timeout` to 5 in the iSCSI configuration file.

4. Enable the iSCSI service:

   ```
   $systemctl enable iscsid
   ```

5. Start the iSCSI service:

   ```
   $systemctl start iscsid
   ```

6. Verify that the iSCSI service is running:

   ```
   $systemctl status iscsid
   ```

Show example

```
● iscsid.service - Open-iSCSI
     Loaded: loaded (/usr/lib/systemd/system/iscsid.service;
enabled; preset: disabled)
     Active: active (running) since Tue 2025-12-02 11:36:21 EST; 2
weeks 1 day ago
TriggeredBy: ● iscsid.socket
       Docs: man:iscsid(8)
             man:iscsiuio(8)
             man:iscsiadm(8)
   Main PID: 2263 (iscsid)
     Status: "Ready to process requests"
      Tasks: 1 (limit: 816061)
     Memory: 18.5M
        CPU: 14.480s
     CGroup: /system.slice/iscsid.service
             └─2263 /usr/sbin/iscsid -f -d2
```

7. Discover the iSCSI targets:

```
$iscsiadm --mode discovery --op update --type sendtargets --portal
<target_IP>
```

show example

```
iscsiadm --mode discovery --op update --type sendtargets --portal
192.168.30.87
192.168.30.87:3260,1139 iqn.1992-
08.com.netapp:sn.064a9b19b3ee11f09dcad039eabac370:vs.23
192.168.31.97:3260,1142 iqn.1992-
08.com.netapp:sn.064a9b19b3ee11f09dcad039eabac370:vs.23
192.168.31.87:3260,1141 iqn.1992-
08.com.netapp:sn.064a9b19b3ee11f09dcad039eabac370:vs.23
192.168.30.97:3260,1140 iqn.1992-
08.com.netapp:sn.064a9b19b3ee11f09dcad039eabac370:vs.23
```

8. Log in to the targets:

```
$iscsiadm --mode node -l all
```

9. Set iSCSI to log in automatically when the host boots:

```
$iscsiadm --mode node -T <target_name> -p <ip:port> -o update -n
node.startup -v automatic
```

You should see an output similar to the following example:

```
iscsiadm --mode node -T iqn.1992-
08.com.netapp:sn.064a9b19b3ee11f09dcad039eabac370:vs.23 -p
192.168.30.87:3260 -o update -n node.startup -v automatic
```

10. Verify the iSCSI sessions:

```
$iscsiadm --mode session
```

Show example

```
iscsiadm --mode session
tcp: [1] 192.168.30.87:3260,1139 iqn.1992-
08.com.netapp:sn.064a9b19b3ee11f09dcad039eabac370:vs.23 (non-flash)
tcp: [2] 192.168.31.97:3260,1142 iqn.1992-
08.com.netapp:sn.064a9b19b3ee11f09dcad039eabac370:vs.23 (non-flash)
tcp: [3] 192.168.31.87:3260,1141 iqn.1992-
08.com.netapp:sn.064a9b19b3ee11f09dcad039eabac370:vs.23 (non-flash)
tcp: [4] 192.168.30.97:3260,1140 iqn.1992-
08.com.netapp:sn.064a9b19b3ee11f09dcad039eabac370:vs.23 (non-flash)
```

Step 5: Optionally, exclude a device from multipathing

If required, you can exclude a device from multipathing by adding the WWID for the unwanted device to the "blacklist" stanza for the `multipath.conf` file.

Steps

1. Determine the WWID:

```
/lib/udev/scsi_id -gud /dev/sda
```

"sda" is the local SCSI disk that you want to add to the blacklist.

An example WWID is `360030057024d0730239134810c0cb833`.

2. Add the WWID to the "blacklist" stanza:

```
blacklist {
         wwid   360030057024d0730239134810c0cb833
        devnode "^(ram|raw|loop|fd|md|dm-|sr|scd|st)[0-9]*"
        devnode "^hd[a-z]"
        devnode "^cciss.*"
}
```

Step 6: Customize multipath parameters for ONTAP LUNs

If your host is connected to LUNs from other vendors and any of the multipath parameter settings are overridden, you need to correct them by adding stanzas later in the `multipath.conf` file that apply specifically to ONTAP LUNs. If you don't do this, the ONTAP LUNs might not work as expected.

Check your `/etc/multipath.conf` file, especially in the defaults section, for settings that might be overriding the default settings for multipath parameters.

> You shouldn't override the recommended parameter settings for ONTAP LUNs. These settings are required for optimal performance of your host configuration. Contact NetApp support, your OS vendor, or both for more information.

The following example shows how to correct an overridden default. In this example, the `multipath.conf` file defines values for `path_checker` and `no_path_retry` that aren't compatible with ONTAP LUNs, and you can't remove these parameters because ONTAP storage arrays are still attached to the host. Instead, you correct the values for `path_checker` and `no_path_retry` by adding a device stanza to the `multipath.conf` file that applies specifically to the ONTAP LUNs.

Show example

```
defaults {
   path_checker       readsector0
   no_path_retry      fail
}

devices {
   device {
      vendor           "NETAPP"
      product          "LUN"
      no_path_retry    queue
      path_checker     tur
   }
}
```

Step 7: Review the known issues

There are no known issues.

What's next?

- Learn about using the Linux Host Utilities tool.
- Learn about ASM mirroring

  Automatic Storage Management (ASM) mirroring might require changes to the Linux multipath settings to allow ASM to recognize a problem and switch over to an alternate failure group. Most ASM configurations on ONTAP use external redundancy, which means that data protection is provided by the external array and ASM doesn't mirror data. Some sites use ASM with normal redundancy to provide two-way mirroring, normally across different sites. See Oracle databases on ONTAP for further information.

- Learn about Oracle Linux KVM and Virtualization

  Oracle Linux can serve as a KVM host. This enables you to run multiple virtual machines on a single physical server using the Linux Kernel-based Virtual Machine (KVM) technology. You can manage and support multiple Oracle Linux KVM hosts by using Oracle Linux Virtualization Manager, which is built from the open-source oVirt project. The KVM host doesn't require explicit host configuration settings for ONTAP LUNs.

Proxmox

Configure Proxmox VE 9.x for FCP and iSCSI with ONTAP storage

Configure Proxmox VE 9.x for multipathing and with specific parameters and settings for FCP and iSCSI protocol operations with ONTAP storage.

FCP and iSCSI with Proxmox VE 9.x have the following known limitations:

- The Linux Host Utilities do not support Proxmox VE 9.x operating systems.
- The SAN boot configuration is not supported.

Step 1: Confirm the multipath configuration for your host

You can use multipathing with Proxmox VE 9.x to manage ONTAP LUNs.

To ensure that multipathing is configured correctly for your host, verify that the `/etc/multipath.conf` file is defined and that you have the NetApp recommended settings configured for your ONTAP LUNs.

Steps

1. Verify that the `/etc/multipath.conf` file exits. If the file doesn't exist, create an empty, zero-byte file:

   ```
   touch /etc/multipath.conf
   ```

2. The first time the `multipath.conf` file is created, you might need to enable and start the multipath services to load the recommended settings:

```
systemctl enable multipathd
```

```
systemctl start multipathd
```

3. Each time you boot the host, the empty `/etc/multipath.conf` zero-byte file automatically loads the NetApp recommended host multipath parameters as the default settings. You shouldn't need to make changes to the `/etc/multipath.conf` file for your host because the operating system is compiled with the multipath parameters that recognize and manage ONTAP LUNs correctly.

   The following table shows the Linux OS native compiled multipath parameter settings for ONTAP LUNs.

   Show parameter settings

Parameter	Setting
detect_prio	yes
dev_loss_tmo	"infinity"
failback	immediate
fast_io_fail_tmo	5
features	"2 pg_init_retries 50"
flush_on_last_del	"always"
hardware_handler	"1"
no_path_retry	queue
path_checker	"tur"
path_grouping_policy	"group_by_prio"
path_selector	"service-time 0"
polling_interval	5
prio	"ontap"
product	LUN
retain_attached_hw_handler	yes
rr_weight	"uniform"
user_friendly_names	no
vendor	NETAPP

4. Optionally, override the default value for the `find_multipaths` parameter to ensure that ONTAP LUNs are correctly discovered and managed by multipathd:

   a. Set `find_multipaths` to "no" in the defaults section of `/etc/multipath.conf`:

```
defaults {
    find_multipaths "no"
}
```

b. Reload the multipath service:

```
systemctl reload multipathd
```

> By default, the Proxmox OS-native multipath configuration sets `find_multipaths` to "**strict**" with the empty zero byte `/etc/multipath.conf` configuration file each time you boot the host. This can prevent the host discovering newly presented ONTAP LUNs as multipath devices, which means they do not appear under multipath control automatically. Existing ONTAP LUNs remain discovered and under multipath control after each reboot.

5. Verify the parameter settings and path status for your ONTAP LUNs:

```
multipath -ll
```

The default multipath parameters support ASA, AFF, and FAS configurations. In these configurations, a single ONTAP LUN shouldn’t require more than four paths. If there are more than four paths, it might cause issues with the paths during a storage failure.

The following example outputs show the correct parameter settings and path status for ONTAP LUNs in an ASA, AFF, or FAS configuration.

ASA configuration

An ASA configuration optimizes all paths to a given LUN, keeping them active. This improves performance by serving I/O operations through all paths at the same time.

Show example

```
multipath -ll
3600a098038315071592b59713261566d dm-38 NETAPP,LUN C-Mode
size=100G features='3 queue_if_no_path pg_init_retries 50'
hwhandler='1 alua' wp=rw
`-+- policy='service-time 0' prio=50 status=active
  |- 8:0:0:7 sdbv 68:144 active ready running
  |- 9:0:0:7 sdbx 68:176 active ready running
  |- 6:0:0:7 sdbr 68:80  active ready running
  `- 7:0:0:7 sdbt 68:112 active ready running
```

AFF or FAS configuration

An AFF or FAS configuration should have two groups of paths with higher and lower priorities. Higher priority Active/Optimized paths are served by the controller where the aggregate is located. Lower priority paths are active but non-optimized because they are served by a different controller. Non-optimized paths are only used when optimized paths aren’t available.

The following example shows the output for an ONTAP LUN with two Active/Optimized paths and two Active/Non-Optimized paths:

Show example

```
multipath -ll
3600a0980383149764b5d567257516273 dm-0 NETAPP,LUN C-Mode
size=150G features='3 queue_if_no_path pg_init_retries 50'
hwhandler='1 alua' wp=rw
|-+- policy='service-time 0' prio=50 status=active
| |- 16:0:3:0  sdcg 69:64  active ready running
| `- 10:0:0:0  sdb  8:16   active ready running
`-+- policy='service-time 0' prio=10 status=enabled
  |- 10:0:1:0  sdc  8:32   active ready running
  `- 16:0:2:0  sdcf 69:48  active ready running
```

Step 2: Confirm the iSCSI configuration for your host

Ensure that iSCSI is configured correctly for your host.

About this task

You perform the following steps on the iSCSI host.

Steps

1. Verify that the iSCSI initiator package (open-iscsi) is installed:

   ```
   $apt list |grep open-iscsi
   ```

   You should see an output similar to the following example:

   ```
   open-iscsi/noble-updates,noble-updates,now 2.1.9-3ubuntu5.4 amd64
   ```

2. Verify the iSCSI initiator node name, which is located in the `/etc/iscsi/initiatorname.iscsi` file:

   ```
   InitiatorName=iqn.YYYY-MM.com.<vendor>:<host_name>
   ```

3. Configure the iSCSI session timeout parameter located in the `/etc/iscsi/iscsid.conf` file:

   ```
   node.session.timeo.replacement_timeout = 5
   ```

   The iSCSI `replacement_timeout` parameter controls how long the iSCSI layer should wait for a timed-out path or session to reestablish itself before failing any commands on it. You should set the value of `replacement_timeout` to 5 in the iSCSI configuration file.

4. Enable the iSCSI service:

   ```
   $systemctl enable iscsid
   ```

5. Start the iSCSI service:

   ```
   $systemctl start iscsid
   ```

6. Verify that the iSCSI service is running:

   ```
   $systemctl status iscsid
   ```

Show example

```
●iscsid.service - iSCSI initiator daemon (iscsid)
     Loaded: loaded (/usr/lib/systemd/system/iscsid.service;
enabled; preset: disabled)
     Active: active (running) since Mon 2026-01-12 12:53:18 IST; 2
days ago
TriggeredBy: ● iscsid.socket
       Docs: man:iscsid(8)
   Main PID: 1127419 (iscsid)
      Tasks: 2 (limit: 76557)
     Memory: 4.3M (peak: 8.8M)
        CPU: 1.657s
     CGroup: /system.slice/iscsid.service
             ├─1127418 /usr/sbin/iscsid
             └─1127419 /usr/sbin/iscsid
```

7. Discover the iSCSI targets:

```
$iscsiadm --mode discovery --op update --type sendtargets --portal
<target_IP>
```

show example

```
iscsiadm --mode discovery --op update --type sendtargets --portal
192.168.100.197
192.168.100.197:3260,1046 iqn.1992-
08.com.netapp:sn.7cd154a7d35411f0a25ed039eaa95f59:vs.8
192.168.200.199:3260,1049 iqn.1992-
08.com.netapp:sn.7cd154a7d35411f0a25ed039eaa95f59:vs.8
192.168.100.199:3260,1048 iqn.1992-
08.com.netapp:sn.7cd154a7d35411f0a25ed039eaa95f59:vs.8
192.168.200.197:3260,1047 iqn.1992-
08.com.netapp:sn.7cd154a7d35411f0a25ed039eaa95f59:vs.8
```

8. Log in to the targets:

```
$iscsiadm --mode node -l all
```

9. Set iSCSI to log in automatically when the host boots:

```
$iscsiadm --mode node -T <target_name> -p <ip:port> -o update -n
node.startup -v automatic
```

You should see an output similar to the following example:

```
iscsiadm --mode node -T iqn.1992-
08.com.netapp:sn.7cd154a7d35411f0a25ed039eaa95f59:vs.8 -p
192.168.100.197:3260 -o update -n node.startup -v automatic
```

10. Verify the iSCSI sessions:

```
$iscsiadm --mode session
```

Show example

```
iscsiadm --mode session
tcp: [1] 192.168.200.197:3260,1047 iqn.1992-
08.com.netapp:sn.7cd154a7d35411f0a25ed039eaa95f59:vs.8 (non-flash)
tcp: [2] 192.168.100.197:3260,1046 iqn.1992-
08.com.netapp:sn.7cd154a7d35411f0a25ed039eaa95f59:vs.8 (non-flash)
tcp: [3] 192.168.100.199:3260,1048 iqn.1992-
08.com.netapp:sn.7cd154a7d35411f0a25ed039eaa95f59:vs.8 (non-flash)
tcp: [4] 192.168.200.199:3260,1049 iqn.1992-
08.com.netapp:sn.7cd154a7d35411f0a25ed039eaa95f59:vs.8 (non-flash)
```

Step 3: Optionally, exclude a device from multipathing

If required, you can exclude a device from multipathing by adding the WWID for the unwanted device to the "blacklist" stanza for the `multipath.conf` file.

Steps

1. Determine the WWID:

```
/lib/udev/scsi_id -gud /dev/sda
```

"sda" is the local SCSI disk that you want to add to the blacklist.

An example WWID is `360030057024d0730239134810c0cb833`.

2. Add the WWID to the "blacklist" stanza:

```
blacklist {
         wwid   360030057024d0730239134810c0cb833
        devnode "^(ram|raw|loop|fd|md|dm-|sr|scd|st)[0-9]*"
        devnode "^hd[a-z]"
        devnode "^cciss.*"
}
```

Step 4: Customize multipath parameters for ONTAP LUNs

If your host is connected to LUNs from other vendors and any of the multipath parameter settings are overridden, you need to correct them by adding stanzas later in the `multipath.conf` file that apply specifically to ONTAP LUNs. If you don't do this, the ONTAP LUNs might not work as expected.

Check your `/etc/multipath.conf` file, especially in the defaults section, for settings that might be overriding the default settings for multipath parameters.

> You shouldn't override the recommended parameter settings for ONTAP LUNs. These settings are required for optimal performance of your host configuration. Contact NetApp support, your OS vendor, or both for more information.

The following example shows how to correct an overridden default. In this example, the `multipath.conf` file defines values for `path_checker` and `no_path_retry` that aren't compatible with ONTAP LUNs, and you can't remove these parameters because ONTAP storage arrays are still attached to the host. Instead, you correct the values for `path_checker` and `no_path_retry` by adding a device stanza to the `multipath.conf` file that applies specifically to the ONTAP LUNs.

Show example

```
defaults {
   path_checker       readsector0
   no_path_retry      fail
}

devices {
   device {
      vendor           "NETAPP"
      product          "LUN"
      no_path_retry    queue
      path_checker     tur
   }
}
```

Step 5: Review the known issues

There are no known issues.

Configure Proxmox VE 8.x for FCP and iSCSI with ONTAP storage

Configure Proxmox VE 8.x for multipathing and with specific parameters and settings for FCP and iSCSI protocol operations with ONTAP storage.

FCP and iSCSI with Proxmox VE 8.x have the following known limitations:

- The Linux Host Utilities do not support Proxmox VE 8.x operating systems.
- The SAN boot configuration is not supported.

Step 1: Confirm the multipath configuration for your host

You can use multipathing with Proxmox VE 8.x to manage ONTAP LUNs.

To ensure that multipathing is configured correctly for your host, verify that the `/etc/multipath.conf` file is defined and that you have the NetApp recommended settings configured for your ONTAP LUNs.

Steps

1. Verify that the `/etc/multipath.conf` file exits. If the file doesn't exist, create an empty, zero-byte file:

```
touch /etc/multipath.conf
```

2. The first time the `multipath.conf` file is created, you might need to enable and start the multipath services to load the recommended settings:

```
systemctl enable multipathd
```

```
systemctl start multipathd
```

3. Each time you boot the host, the empty `/etc/multipath.conf` zero-byte file automatically loads the NetApp recommended host multipath parameters as the default settings. You shouldn't need to make changes to the `/etc/multipath.conf` file for your host because the operating system is compiled with the multipath parameters that recognize and manage ONTAP LUNs correctly.

   The following table shows the Linux OS native compiled multipath parameter settings for ONTAP LUNs.

Show parameter settings

Parameter	Setting
detect_prio	yes
dev_loss_tmo	"infinity"
failback	immediate
fast_io_fail_tmo	5
features	"2 pg_init_retries 50"
flush_on_last_del	"always"
hardware_handler	"1"
no_path_retry	queue
path_checker	"tur"
path_grouping_policy	"group_by_prio"
path_selector	"service-time 0"
polling_interval	5
prio	"ontap"
product	LUN
retain_attached_hw_handler	yes
rr_weight	"uniform"
user_friendly_names	no
vendor	NETAPP

4. Optionally, override the default value for the `find_multipaths` parameter to ensure that ONTAP LUNs are correctly discovered and managed by multipathd:

   a. Set `find_multipaths` to "no" in the defaults section of `/etc/multipath.conf`:

   ```
   defaults {
       find_multipaths "no"
   }
   ```

   b. Reload the multipath service:

   ```
   systemctl reload multipathd
   ```

By default, the Proxmox OS-native multipath configuration sets `find_multipaths` to "**strict**" with the empty zero byte `/etc/multipath.conf` configuration file each time you boot the host. This can prevent the host discovering newly presented ONTAP LUNs as multipath devices, which means they do not appear under multipath control automatically. Existing ONTAP LUNs remain discovered and under multipath control after each reboot.

5. Verify the parameter settings and path status for your ONTAP LUNs:

```
multipath -ll
```

The default multipath parameters support ASA, AFF, and FAS configurations. In these configurations, a single ONTAP LUN shouldn’t require more than four paths. If there are more than four paths, it might cause issues with the paths during a storage failure.

The following example outputs show the correct parameter settings and path status for ONTAP LUNs in an ASA, AFF, or FAS configuration.

ASA configuration

An ASA configuration optimizes all paths to a given LUN, keeping them active. This improves performance by serving I/O operations through all paths at the same time.

Show example

```
multipath -ll
3600a098038315071592b59713261566d dm-38 NETAPP,LUN C-Mode
size=100G features='3 queue_if_no_path pg_init_retries 50'
hwhandler='1 alua' wp=rw
`-+- policy='service-time 0' prio=50 status=active
  |- 8:0:0:7 sdbv 68:144 active ready running
  |- 9:0:0:7 sdbx 68:176 active ready running
  |- 6:0:0:7 sdbr 68:80  active ready running
  `- 7:0:0:7 sdbt 68:112 active ready running
```

AFF or FAS configuration

An AFF or FAS configuration should have two groups of paths with higher and lower priorities. Higher priority Active/Optimized paths are served by the controller where the aggregate is located. Lower priority paths are active but non-optimized because they are served by a different controller. Non-optimized paths are only used when optimized paths aren’t available.

The following example shows the output for an ONTAP LUN with two Active/Optimized paths and two Active/Non-Optimized paths:

Show example

```
multipath -ll
3600a0980383149764b5d567257516273 dm-0 NETAPP,LUN C-Mode
size=150G features='3 queue_if_no_path pg_init_retries 50'
hwhandler='1 alua' wp=rw
|-+- policy='service-time 0' prio=50 status=active
| |- 16:0:3:0  sdcg 69:64  active ready running
| `- 10:0:0:0  sdb  8:16   active ready running
`-+- policy='service-time 0' prio=10 status=enabled
  |- 10:0:1:0  sdc  8:32   active ready running
  `- 16:0:2:0  sdcf 69:48  active ready running
```

Step 2: Confirm the iSCSI configuration for your host

Ensure that iSCSI is configured correctly for your host.

About this task

You perform the following steps on the iSCSI host.

Steps

1. Verify that the iSCSI initiator package (open-iscsi) is installed:

```
$apt list |grep open-iscsi
```

   You should see an output similar to the following example:

```
open-iscsi/noble-updates,noble-updates,now 2.1.9-3ubuntu5.4 amd64
```

2. Verify the iSCSI initiator node name, which is located in the `/etc/iscsi/initiatorname.iscsi` file:

```
InitiatorName=iqn.YYYY-MM.com.<vendor>:<host_name>
```

3. Configure the iSCSI session timeout parameter located in the `/etc/iscsi/iscsid.conf` file:

```
node.session.timeo.replacement_timeout = 5
```

   The iSCSI `replacement_timeout` parameter controls how long the iSCSI layer should wait for a timed-out path or session to reestablish itself before failing any commands on it. You should set the value of `replacement_timeout` to 5 in the iSCSI configuration file.

4. Enable the iSCSI service:

```
$systemctl enable iscsid
```

5. Start the iSCSI service:

```
$systemctl start iscsid
```

6. Verify that the iSCSI service is running:

```
$systemctl status iscsid
```

Show example

```
●iscsid.service - iSCSI initiator daemon (iscsid)
     Loaded: loaded (/usr/lib/systemd/system/iscsid.service;
enabled; preset: disabled)
     Active: active (running) since Mon 2026-01-12 12:53:18 IST; 2
days ago
TriggeredBy: ● iscsid.socket
       Docs: man:iscsid(8)
   Main PID: 1127419 (iscsid)
      Tasks: 2 (limit: 76557)
     Memory: 4.3M (peak: 8.8M)
        CPU: 1.657s
     CGroup: /system.slice/iscsid.service
             ├─1127418 /usr/sbin/iscsid
             └─1127419 /usr/sbin/iscsid
```

7. Discover the iSCSI targets:

```
$iscsiadm --mode discovery --op update --type sendtargets --portal
<target_IP>
```

show example

```
iscsiadm --mode discovery --op update --type sendtargets --portal
192.168.100.197
192.168.100.197:3260,1046 iqn.1992-
08.com.netapp:sn.7cd154a7d35411f0a25ed039eaa95f59:vs.8
192.168.200.199:3260,1049 iqn.1992-
08.com.netapp:sn.7cd154a7d35411f0a25ed039eaa95f59:vs.8
192.168.100.199:3260,1048 iqn.1992-
08.com.netapp:sn.7cd154a7d35411f0a25ed039eaa95f59:vs.8
192.168.200.197:3260,1047 iqn.1992-
08.com.netapp:sn.7cd154a7d35411f0a25ed039eaa95f59:vs.8
```

8. Log in to the targets:

```
$iscsiadm --mode node -l all
```

9. Set iSCSI to log in automatically when the host boots:

```
$iscsiadm --mode node -T <target_name> -p <ip:port> -o update -n
node.startup -v automatic
```

You should see an output similar to the following example:

```
iscsiadm --mode node -T iqn.1992-
08.com.netapp:sn.7cd154a7d35411f0a25ed039eaa95f59:vs.8 -p
192.168.100.197:3260 -o update -n node.startup -v automatic
```

10. Verify the iSCSI sessions:

```
$iscsiadm --mode session
```

Show example

```
iscsiadm --mode session
tcp: [1] 192.168.200.197:3260,1047 iqn.1992-
08.com.netapp:sn.7cd154a7d35411f0a25ed039eaa95f59:vs.8 (non-flash)
tcp: [2] 192.168.100.197:3260,1046 iqn.1992-
08.com.netapp:sn.7cd154a7d35411f0a25ed039eaa95f59:vs.8 (non-flash)
tcp: [3] 192.168.100.199:3260,1048 iqn.1992-
08.com.netapp:sn.7cd154a7d35411f0a25ed039eaa95f59:vs.8 (non-flash)
tcp: [4] 192.168.200.199:3260,1049 iqn.1992-
08.com.netapp:sn.7cd154a7d35411f0a25ed039eaa95f59:vs.8 (non-flash)
```

Step 3: Optionally, exclude a device from multipathing

If required, you can exclude a device from multipathing by adding the WWID for the unwanted device to the "blacklist" stanza for the `multipath.conf` file.

Steps

1. Determine the WWID:

```
/lib/udev/scsi_id -gud /dev/sda
```

"sda" is the local SCSI disk that you want to add to the blacklist.

An example WWID is `360030057024d0730239134810c0cb833`.

2. Add the WWID to the "blacklist" stanza:

```
blacklist {
         wwid   360030057024d0730239134810c0cb833
        devnode "^(ram|raw|loop|fd|md|dm-|sr|scd|st)[0-9]*"
        devnode "^hd[a-z]"
        devnode "^cciss.*"
}
```

Step 4: Customize multipath parameters for ONTAP LUNs

If your host is connected to LUNs from other vendors and any of the multipath parameter settings are overridden, you need to correct them by adding stanzas later in the `multipath.conf` file that apply specifically to ONTAP LUNs. If you don't do this, the ONTAP LUNs might not work as expected.

Check your `/etc/multipath.conf` file, especially in the defaults section, for settings that might be overriding the default settings for multipath parameters.

> ⚠ You shouldn't override the recommended parameter settings for ONTAP LUNs. These settings are required for optimal performance of your host configuration. Contact NetApp support, your OS vendor, or both for more information.

The following example shows how to correct an overridden default. In this example, the `multipath.conf` file defines values for `path_checker` and `no_path_retry` that aren't compatible with ONTAP LUNs, and you can't remove these parameters because ONTAP storage arrays are still attached to the host. Instead, you correct the values for `path_checker` and `no_path_retry` by adding a device stanza to the `multipath.conf` file that applies specifically to the ONTAP LUNs.

Show example

```
defaults {
   path_checker       readsector0
   no_path_retry      fail
}

devices {
   device {
      vendor           "NETAPP"
      product          "LUN"
      no_path_retry    queue
      path_checker     tur
   }
}
```

Step 5: Review the known issues

There are no known issues.

RHEL

Configure RHEL 10.x for FCP and iSCSI with ONTAP storage

The Linux Host Utilities software provides management and diagnostic tools for Linux hosts that are connected to ONTAP storage. When you install the Linux Host Utilities on a Red Hat Enterprise Linux (RHEL) 10.x host, you can use the Host Utilities to help you manage FCP and iSCSI protocol operations with ONTAP LUNs.

Step 1: Optionally, enable SAN booting

You can configure your host to use SAN booting to simplify deployment and improve scalability.

Before you begin

Use the Interoperability Matrix Tool to verify that your Linux OS, host bus adapter (HBA), HBA firmware, HBA boot BIOS, and ONTAP version support SAN booting.

Steps

1. Create a SAN boot LUN and map it to the host.
2. Enable SAN booting in the server BIOS for the ports to which the SAN boot LUN is mapped.

   For information on how to enable the HBA BIOS, see your vendor-specific documentation.

3. Verify that the configuration was successful by rebooting the host and verifying that the OS is up and running.

Step 2: Install the Linux Host Utilities

NetApp strongly recommends installing the Linux Host Utilities to support ONTAP LUN management and assist technical support with gathering configuration data.

Install Linux Host Utilities 8.0.

Installing the Linux Host Utilities doesn't change any host timeout settings on your Linux host.

Step 3: Confirm the multipath configuration for your host

You can use multipathing with RHEL 10.x to manage ONTAP LUNs.

To ensure that multipathing is configured correctly for your host, verify that the `/etc/multipath.conf` file is defined and that you have the NetApp recommended settings configured for your ONTAP LUNs.

Steps

1. Verify that the `/etc/multipath.conf` file exits. If the file doesn't exist, create an empty, zero-byte file:

   ```
   touch /etc/multipath.conf
   ```

2. The first time the `multipath.conf` file is created, you might need to enable and start the multipath services to load the recommended settings:

```
systemctl enable multipathd
```

```
systemctl start multipathd
```

3. Each time you boot the host, the empty `/etc/multipath.conf` zero-byte file automatically loads the NetApp recommended host multipath parameters as the default settings. You shouldn't need to make changes to the `/etc/multipath.conf` file for your host because the operating system is compiled with the multipath parameters that recognize and manage ONTAP LUNs correctly.

   The following table shows the Linux OS native compiled multipath parameter settings for ONTAP LUNs.

   Show parameter settings

Parameter	Setting
detect_prio	yes
dev_loss_tmo	"infinity"
failback	immediate
fast_io_fail_tmo	5
features	"2 pg_init_retries 50"
flush_on_last_del	"yes"
hardware_handler	"0"
no_path_retry	queue
path_checker	"tur"
path_grouping_policy	"group_by_prio"
path_selector	"service-time 0"
polling_interval	5
prio	"ontap"
product	LUN
retain_attached_hw_handler	yes
rr_weight	"uniform"
user_friendly_names	no
vendor	NETAPP

4. Verify the parameter settings and path status for your ONTAP LUNs:

```
multipath -ll
```

The default multipath parameters support ASA, AFF, and FAS configurations. In these configurations, a single ONTAP LUN shouldn’t require more than four paths. Having more than four paths can cause problems during a storage failure.

The following example outputs show the correct parameter settings and path status for ONTAP LUNs in an ASA, AFF, or FAS configuration.

ASA configuration

An ASA configuration optimizes all paths to a given LUN, keeping them active. This improves performance by serving I/O operations through all paths at the same time.

Show example

```
# multipath -ll
3600a098038314e535a24584e4b496252 dm-32 NETAPP,LUN C-Mode
size=10G features='3 queue_if_no_path pg_init_retries 50'
hwhandler='1 alua' wp=rw
`-+- policy='service-time 0' prio=50 status=active
  |- 11:0:0:41 sdan 66:112  active ready running
  |- 11:0:1:41 sdcb 68:240  active ready running
  |- 14:0:2:41 sdfd 129:240 active ready running
  `- 14:0:0:41 sddp 71:112  active ready running
```

AFF or FAS configuration

An AFF or FAS configuration should have two groups of paths with higher and lower priorities. Higher priority Active/Optimized paths are served by the controller where the aggregate is located. Lower priority paths are active but non-optimized because they are served by a different controller. Non-optimized paths are only used when optimized paths aren’t available.

The following example displays the output for an ONTAP LUN with two Active/Optimized paths and two Active/Non-Optimized paths:

Show example

```
# multipath -ll
3600a0980383149764b5d567257516273 dm-0 NETAPP,LUN C-Mode
size=150G features='3 queue_if_no_path pg_init_retries 50'
hwhandler='1 alua' wp=rw
|-+- policy='service-time 0' prio=50 status=active
| |- 16:0:3:0  sdcg 69:64  active ready running
| `- 10:0:0:0  sdb  8:16   active ready running
`-+- policy='service-time 0' prio=10 status=enabled
  |- 10:0:1:0  sdc  8:32   active ready running
  `- 16:0:2:0  sdcf 69:48  active ready running
```

Step 4: Confirm the iSCSI configuration for your host

Ensure that iSCSI is configured correctly for your host.

About this task

You perform the following steps on the iSCSI host.

Steps

1. Verify that the iSCSI initiator package (iscsi-initiator-utils) is installed:

   ```
   rpm -qa | grep iscsi-initiator-utils
   ```

   You should see an output similar to the following example:

   ```
   iscsi-initiator-utils-6.2.1.11-0.git4b3e853.el9.x86_64
   ```

2. Verify the iSCSI initiator node name, which is located in the `/etc/iscsi/initiatorname.iscsi` file:

   ```
   InitiatorName=iqn.YYYY-MM.com.<vendor>:<host_name>
   ```

3. Configure the iSCSI session timeout parameter located in the `/etc/iscsi/iscsid.conf` file:

   ```
   node.session.timeo.replacement_timeout = 5
   ```

   The iSCSI `replacement_timeout` parameter controls how long the iSCSI layer should wait for a timed-out path or session to reestablish itself before failing any commands on it. You should set the value of `replacement_timeout` to 5 in the iSCSI configuration file.

4. Enable the iSCSI service:

   ```
   $systemctl enable iscsid
   ```

5. Start the iSCSI service:

   ```
   $systemctl start iscsid
   ```

6. Verify that the iSCSI service is running:

   ```
   $systemctl status iscsid
   ```

Show example

```
● iscsid.service - Open-iSCSI
     Loaded: loaded (/usr/lib/systemd/system/iscsid.service;
enabled; preset: disabled)
     Active: active (running) since Tue 2025-12-02 11:36:21 EST; 2
weeks 1 day ago
TriggeredBy: ● iscsid.socket
       Docs: man:iscsid(8)
             man:iscsiuio(8)
             man:iscsiadm(8)
   Main PID: 2263 (iscsid)
     Status: "Ready to process requests"
      Tasks: 1 (limit: 816061)
     Memory: 18.5M
        CPU: 14.480s
     CGroup: /system.slice/iscsid.service
             └─2263 /usr/sbin/iscsid -f -d2
```

7. Discover the iSCSI targets:

```
$iscsiadm --mode discovery --op update --type sendtargets --portal
<target_IP>
```

show example

```
iscsiadm --mode discovery --op update --type sendtargets --portal
192.168.30.87
192.168.30.87:3260,1139 iqn.1992-
08.com.netapp:sn.064a9b19b3ee11f09dcad039eabac370:vs.23
192.168.31.97:3260,1142 iqn.1992-
08.com.netapp:sn.064a9b19b3ee11f09dcad039eabac370:vs.23
192.168.31.87:3260,1141 iqn.1992-
08.com.netapp:sn.064a9b19b3ee11f09dcad039eabac370:vs.23
192.168.30.97:3260,1140 iqn.1992-
08.com.netapp:sn.064a9b19b3ee11f09dcad039eabac370:vs.23
```

8. Log in to the targets:

```
$iscsiadm --mode node -l all
```

9. Set iSCSI to log in automatically when the host boots:

```
$iscsiadm --mode node -T <target_name> -p <ip:port> -o update -n
node.startup -v automatic
```

You should see an output similar to the following example:

```
iscsiadm --mode node -T iqn.1992-
08.com.netapp:sn.064a9b19b3ee11f09dcad039eabac370:vs.23 -p
192.168.30.87:3260 -o update -n node.startup -v automatic
```

10. Verify the iSCSI sessions:

```
$iscsiadm --mode session
```

Show example

```
iscsiadm --mode session
tcp: [1] 192.168.30.87:3260,1139 iqn.1992-
08.com.netapp:sn.064a9b19b3ee11f09dcad039eabac370:vs.23 (non-flash)
tcp: [2] 192.168.31.97:3260,1142 iqn.1992-
08.com.netapp:sn.064a9b19b3ee11f09dcad039eabac370:vs.23 (non-flash)
tcp: [3] 192.168.31.87:3260,1141 iqn.1992-
08.com.netapp:sn.064a9b19b3ee11f09dcad039eabac370:vs.23 (non-flash)
tcp: [4] 192.168.30.97:3260,1140 iqn.1992-
08.com.netapp:sn.064a9b19b3ee11f09dcad039eabac370:vs.23 (non-flash)
```

Step 5: Optionally, exclude a device from multipathing

If required, you can exclude a device from multipathing by adding the WWID for the unwanted device to the "blacklist" stanza for the `multipath.conf` file.

Steps

1. Determine the WWID:

```
/lib/udev/scsi_id -gud /dev/sda
```

"sda" is the local SCSI disk that you want to add to the blacklist.

An example WWID is `360030057024d0730239134810c0cb833`.

2. Add the WWID to the "blacklist" stanza:

```
blacklist {
         wwid   360030057024d0730239134810c0cb833
        devnode "^(ram|raw|loop|fd|md|dm-|sr|scd|st)[0-9]*"
        devnode "^hd[a-z]"
        devnode "^cciss.*"
}
```

Step 6: Customize multipath parameters for ONTAP LUNs

If your host is connected to LUNs from other vendors and any of the multipath parameter settings are overridden, you need to correct them by adding stanzas later in the `multipath.conf` file that apply specifically to ONTAP LUNs. If you don't do this, the ONTAP LUNs might not work as expected.

Check your `/etc/multipath.conf` file, especially in the defaults section, for settings that might be overriding the default settings for multipath parameters.

> You shouldn't override the recommended parameter settings for ONTAP LUNs. These settings are required for optimal performance of your host configuration. Contact NetApp support, your OS vendor, or both for more information.

The following example shows how to correct an overridden default. In this example, the `multipath.conf` file defines values for `path_checker` and `no_path_retry` that aren't compatible with ONTAP LUNs, and you can't remove these parameters because ONTAP storage arrays are still attached to the host. Instead, you correct the values for `path_checker` and `no_path_retry` by adding a device stanza to the `multipath.conf` file that applies specifically to the ONTAP LUNs.

Show example

```
defaults {
   path_checker       readsector0
   no_path_retry      fail
}

devices {
   device {
      vendor           "NETAPP"
      product          "LUN"
      no_path_retry    queue
      path_checker     tur
   }
}
```

Step 7: Review the known issues

There are no known issues.

What's next?

- Learn about using the Linux Host Utilities tool.
- Learn about ASM mirroring

  Automatic Storage Management (ASM) mirroring might require changes to the Linux multipath settings to allow ASM to recognize a problem and switch over to an alternate failure group. Most ASM configurations on ONTAP use external redundancy, which means that data protection is provided by the external array and ASM doesn't mirror data. Some sites use ASM with normal redundancy to provide two-way mirroring, normally across different sites. See Oracle databases on ONTAP for further information.

- Learn about Red Hat Linux Virtualization (KVM)

  Red Hat Linux can serve as a KVM host. This enables you to run multiple virtual machines on a single physical server using the Linux Kernel-based Virtual Machine (KVM) technology. The KVM host doesn't require explicit host configuration settings for ONTAP LUNs.

Configure RHEL 9.x for FCP and iSCSI with ONTAP storage

The Linux Host Utilities software provides management and diagnostic tools for Linux hosts that are connected to ONTAP storage. When you install the Linux Host Utilities on a Red Hat Enterprise Linux (RHEL) 9.x host, you can use the Host Utilities to help you manage FCP and iSCSI protocol operations with ONTAP LUNs.

Step 1: Optionally, enable SAN booting

You can configure your host to use SAN booting to simplify deployment and improve scalability.

Before you begin

Use the Interoperability Matrix Tool to verify that your Linux OS, host bus adapter (HBA), HBA firmware, HBA boot BIOS, and ONTAP version support SAN booting.

Steps

1. Create a SAN boot LUN and map it to the host.
2. Enable SAN booting in the server BIOS for the ports to which the SAN boot LUN is mapped.

   For information on how to enable the HBA BIOS, see your vendor-specific documentation.

3. Verify that the configuration was successful by rebooting the host and verifying that the OS is up and running.

Step 2: Install the Linux Host Utilities

NetApp strongly recommends installing the Linux Host Utilities to support ONTAP LUN management and assist technical support with gathering configuration data.

Install Linux Host Utilities 8.0.

Installing the Linux Host Utilities doesn't change any host timeout settings on your Linux host.

Step 3: Confirm the multipath configuration for your host

You can use multipathing with RHEL 9.x to manage ONTAP LUNs.

To ensure that multipathing is configured correctly for your host, verify that the `/etc/multipath.conf` file is defined and that you have the NetApp recommended settings configured for your ONTAP LUNs.

Steps

1. Verify that the `/etc/multipath.conf` file exits. If the file doesn't exist, create an empty, zero-byte file:

```
touch /etc/multipath.conf
```

2. The first time the `multipath.conf` file is created, you might need to enable and start the multipath services to load the recommended settings:

```
systemctl enable multipathd
```

```
systemctl start multipathd
```

3. Each time you boot the host, the empty `/etc/multipath.conf` zero-byte file automatically loads the NetApp recommended host multipath parameters as the default settings. You shouldn't need to make changes to the `/etc/multipath.conf` file for your host because the operating system is compiled with the multipath parameters that recognize and manage ONTAP LUNs correctly.

   The following table shows the Linux OS native compiled multipath parameter settings for ONTAP LUNs.

Show parameter settings

Parameter	Setting
detect_prio	yes
dev_loss_tmo	"infinity"
failback	immediate
fast_io_fail_tmo	5
features	"2 pg_init_retries 50"
flush_on_last_del	"yes"
hardware_handler	"0"
no_path_retry	queue
path_checker	"tur"
path_grouping_policy	"group_by_prio"
path_selector	"service-time 0"
polling_interval	5
prio	"ontap"
product	LUN
retain_attached_hw_handler	yes
rr_weight	"uniform"
user_friendly_names	no
vendor	NETAPP

4. Verify the parameter settings and path status for your ONTAP LUNs:

```
multipath -ll
```

The default multipath parameters support ASA, AFF, and FAS configurations. In these configurations, a single ONTAP LUN shouldn’t require more than four paths. Having more than four paths can cause problems during a storage failure.

The following example outputs show the correct parameter settings and path status for ONTAP LUNs in an ASA, AFF, or FAS configuration.

ASA configuration

An ASA configuration optimizes all paths to a given LUN, keeping them active. This improves performance by serving I/O operations through all paths at the same time.

Show example

```
multipath -ll
3600a098038314c4a433f577471797958 dm-2 NETAPP,LUN C-Mode
size=180G features='3 queue_if_no_path pg_init_retries 50'
hwhandler='1 alua' wp=rw
`-+- policy='service-time 0' prio=50 status=active
  |- 14:0:0:0  sdc  8:32   active ready running
  |- 17:0:0:0  sdas 66:192 active ready running
  |- 14:0:3:0  sdar 66:176 active ready running
  `- 17:0:3:0  sdch 69:80  active ready running
```

AFF or FAS configuration

An AFF or FAS configuration should have two groups of paths with higher and lower priorities. Higher priority Active/Optimized paths are served by the controller where the aggregate is located. Lower priority paths are active but non-optimized because they are served by a different controller. Non-optimized paths are only used when optimized paths aren’t available.

The following example displays the output for an ONTAP LUN with two Active/Optimized paths and two Active/Non-Optimized paths:

Show example

```
multipath -ll
3600a0980383149764b5d567257516273 dm-0 NETAPP,LUN C-Mode
size=150G features='3 queue_if_no_path pg_init_retries 50'
hwhandler='1 alua' wp=rw
|-+- policy='service-time 0' prio=50 status=active
| |- 16:0:3:0  sdcg 69:64  active ready running
| `- 10:0:0:0  sdb  8:16   active ready running
`-+- policy='service-time 0' prio=10 status=enabled
  |- 10:0:1:0  sdc  8:32   active ready running
  `- 16:0:2:0  sdcf 69:48  active ready running
```

Step 4: Confirm the iSCSI configuration for your host

Ensure that iSCSI is configured correctly for your host.

About this task

You perform the following steps on the iSCSI host.

Steps

1. Verify that the iSCSI initiator package (iscsi-initiator-utils) is installed:

   ```
   rpm -qa | grep iscsi-initiator-utils
   ```

   You should see an output similar to the following example:

   ```
   iscsi-initiator-utils-6.2.1.11-0.git4b3e853.el9.x86_64
   ```

2. Verify the iSCSI initiator node name, which is located in the `/etc/iscsi/initiatorname.iscsi` file:

   ```
   InitiatorName=iqn.YYYY-MM.com.<vendor>:<host_name>
   ```

3. Configure the iSCSI session timeout parameter located in the `/etc/iscsi/iscsid.conf` file:

   ```
   node.session.timeo.replacement_timeout = 5
   ```

   The iSCSI `replacement_timeout` parameter controls how long the iSCSI layer should wait for a timed-out path or session to reestablish itself before failing any commands on it. You should set the value of `replacement_timeout` to 5 in the iSCSI configuration file.

4. Enable the iSCSI service:

   ```
   $systemctl enable iscsid
   ```

5. Start the iSCSI service:

   ```
   $systemctl start iscsid
   ```

6. Verify that the iSCSI service is running:

   ```
   $systemctl status iscsid
   ```

Show example

```
● iscsid.service - Open-iSCSI
     Loaded: loaded (/usr/lib/systemd/system/iscsid.service;
enabled; preset: disabled)
     Active: active (running) since Tue 2025-12-02 11:36:21 EST; 2
weeks 1 day ago
TriggeredBy: ● iscsid.socket
       Docs: man:iscsid(8)
             man:iscsiuio(8)
             man:iscsiadm(8)
   Main PID: 2263 (iscsid)
     Status: "Ready to process requests"
      Tasks: 1 (limit: 816061)
     Memory: 18.5M
        CPU: 14.480s
     CGroup: /system.slice/iscsid.service
             └─2263 /usr/sbin/iscsid -f -d2
```

7. Discover the iSCSI targets:

```
$iscsiadm --mode discovery --op update --type sendtargets --portal
<target_IP>
```

show example

```
iscsiadm --mode discovery --op update --type sendtargets --portal
192.168.30.87
192.168.30.87:3260,1139 iqn.1992-
08.com.netapp:sn.064a9b19b3ee11f09dcad039eabac370:vs.23
192.168.31.97:3260,1142 iqn.1992-
08.com.netapp:sn.064a9b19b3ee11f09dcad039eabac370:vs.23
192.168.31.87:3260,1141 iqn.1992-
08.com.netapp:sn.064a9b19b3ee11f09dcad039eabac370:vs.23
192.168.30.97:3260,1140 iqn.1992-
08.com.netapp:sn.064a9b19b3ee11f09dcad039eabac370:vs.23
```

8. Log in to the targets:

```
$iscsiadm --mode node -l all
```

9. Set iSCSI to log in automatically when the host boots:

```
$iscsiadm --mode node -T <target_name> -p <ip:port> -o update -n
node.startup -v automatic
```

You should see an output similar to the following example:

```
iscsiadm --mode node -T iqn.1992-
08.com.netapp:sn.064a9b19b3ee11f09dcad039eabac370:vs.23 -p
192.168.30.87:3260 -o update -n node.startup -v automatic
```

10. Verify the iSCSI sessions:

```
$iscsiadm --mode session
```

Show example

```
iscsiadm --mode session
tcp: [1] 192.168.30.87:3260,1139 iqn.1992-
08.com.netapp:sn.064a9b19b3ee11f09dcad039eabac370:vs.23 (non-flash)
tcp: [2] 192.168.31.97:3260,1142 iqn.1992-
08.com.netapp:sn.064a9b19b3ee11f09dcad039eabac370:vs.23 (non-flash)
tcp: [3] 192.168.31.87:3260,1141 iqn.1992-
08.com.netapp:sn.064a9b19b3ee11f09dcad039eabac370:vs.23 (non-flash)
tcp: [4] 192.168.30.97:3260,1140 iqn.1992-
08.com.netapp:sn.064a9b19b3ee11f09dcad039eabac370:vs.23 (non-flash)
```

Step 5: Optionally, exclude a device from multipathing

If required, you can exclude a device from multipathing by adding the WWID for the unwanted device to the "blacklist" stanza for the `multipath.conf` file.

Steps

1. Determine the WWID:

```
/lib/udev/scsi_id -gud /dev/sda
```

"sda" is the local SCSI disk that you want to add to the blacklist.

An example WWID is `360030057024d0730239134810c0cb833`.

2. Add the WWID to the "blacklist" stanza:

```
blacklist {
         wwid   360030057024d0730239134810c0cb833
        devnode "^(ram|raw|loop|fd|md|dm-|sr|scd|st)[0-9]*"
        devnode "^hd[a-z]"
        devnode "^cciss.*"
}
```

Step 6: Customize multipath parameters for ONTAP LUNs

If your host is connected to LUNs from other vendors and any of the multipath parameter settings are overridden, you need to correct them by adding stanzas later in the `multipath.conf` file that apply specifically to ONTAP LUNs. If you don't do this, the ONTAP LUNs might not work as expected.

Check your `/etc/multipath.conf` file, especially in the defaults section, for settings that might be overriding the default settings for multipath parameters.

> ⚠ You shouldn't override the recommended parameter settings for ONTAP LUNs. These settings are required for optimal performance of your host configuration. Contact NetApp support, your OS vendor, or both for more information.

The following example shows how to correct an overridden default. In this example, the `multipath.conf` file defines values for `path_checker` and `no_path_retry` that aren't compatible with ONTAP LUNs, and you can't remove these parameters because ONTAP storage arrays are still attached to the host. Instead, you correct the values for `path_checker` and `no_path_retry` by adding a device stanza to the `multipath.conf` file that applies specifically to the ONTAP LUNs.

Show example

```
defaults {
   path_checker       readsector0
   no_path_retry      fail
}

devices {
   device {
      vendor           "NETAPP"
      product          "LUN"
      no_path_retry    queue
      path_checker     tur
   }
}
```

Step 7: Review the known issues

RHEL 9.x with ONTAP storage has the following known issues.

9.3

NetApp Bug ID	Title	Description	JIRA ID
1508554	NetApp Linux Host Utilities CLI requires additional library package dependencies to support Emulex host bus adapter (HBA) adapter discovery	In RHEL 9.x, the NetApp Linux SAN Host Utilities CLI `sanlun fcp show adapter -v` fails because the library package dependencies to support Emulex host bus adapter (HBA) discovery cannot be found.	Not applicable
1593771	A Red Hat Enterprise Linux 9.3 QLogic SAN host encounters loss of partial multipaths during storage mobility operations	During the ONTAP storage controller takeover operation, half of the multipaths are expected to go down or switch to a failover mode and then recover to full path count during the giveback workflow. However, with a Red Hat Enterprise Linux (RHEL) 9.3 QLogic host, only partial multipaths are recovered after a storage failover giveback operation.	RHEL 17811

9.2

NetApp Bug ID	Title	Description
1508554	NetApp Linux Host Utilities CLI requires additional library package dependencies to support Emulex HBA adapter discovery	In RHEL 9.2, the NetApp Linux SAN Host Utilities CLI `sanlun fcp show adapter -v` fails because the library package dependencies to support HBA discovery cannot be found.
1537359	A Red Hat Linux 9.2 SAN booted host with Emulex HBA encounters stalled tasks leading to kernel disruption	During a storage failover giveback operation, a Red Hat Linux 9.2 SAN booted host with an Emulex host bus adapeter (HBA) encounters stalled tasks leading to kernel disruption. The kernel disruption causes the operating system to reboot and if `kdump` is configured, it generates the `vmcore` file under the `/var/crash/` directory. The issue is being triaged with the `lpfc` driver but it cannot be reproduced consistently.

9.1

NetApp Bug ID	Title	Description
1508554	NetApp Linux Host Utilities CLI requires additional library package dependencies to support Emulex HBA adapter discovery	In RHEL 9.1, the NetApp Linux SAN Host Utilities CLI `sanlun fcp show adapter -v` fails because the library package dependencies to support HBA discovery cannot be found.

What's next?

- Learn about using the Linux Host Utilities tool.
- Learn about ASM mirroring

  Automatic Storage Management (ASM) mirroring might require changes to the Linux multipath settings to allow ASM to recognize a problem and switch over to an alternate failure group. Most ASM configurations on ONTAP use external redundancy, which means that data protection is provided by the external array and ASM doesn't mirror data. Some sites use ASM with normal redundancy to provide two-way mirroring, normally across different sites. See Oracle databases on ONTAP for further information.

- Learn about Red Hat Linux Virtualization (KVM)

  Red Hat Linux can serve as a KVM host. This enables you to run multiple virtual machines on a single physical server using the Linux Kernel-based Virtual Machine (KVM) technology. The KVM host doesn't require explicit host configuration settings for ONTAP LUNs.

Configure RHEL 8.x for FCP and iSCSI with ONTAP storage

The Linux Host Utilities software provides management and diagnostic tools for Linux hosts that are connected to ONTAP storage. When you install the Linux Host Utilities on a Red Hat Enterprise Linux (RHEL) 8.x host, you can use the Host Utilities to help you manage FCP and iSCSI protocol operations with ONTAP LUNs.

Step 1: Optionally, enable SAN booting

You can configure your host to use SAN booting to simplify deployment and improve scalability.

Before you begin

Use the Interoperability Matrix Tool to verify that your Linux OS, host bus adapter (HBA), HBA firmware, HBA boot BIOS, and ONTAP version support SAN booting.

Steps

1. Create a SAN boot LUN and map it to the host.
2. Enable SAN booting in the server BIOS for the ports to which the SAN boot LUN is mapped.

   For information on how to enable the HBA BIOS, see your vendor-specific documentation.

3. Verify that the configuration was successful by rebooting the host and verifying that the OS is up and running.

Step 2: Install the Linux Host Utilities

NetApp strongly recommends installing the Linux Host Utilities to support ONTAP LUN management and assist technical support with gathering configuration data.

Install Linux Host Utilities 8.0.

Installing the Linux Host Utilities doesn't change any host timeout settings on your Linux host.

Step 3: Confirm the multipath configuration for your host

You can use multipathing with RHEL 8.x to manage ONTAP LUNs.

To ensure that multipathing is configured correctly for your host, verify that the `/etc/multipath.conf` file is defined and that you have the NetApp recommended settings configured for your ONTAP LUNs.

Steps

1. Verify that the `/etc/multipath.conf` file exits. If the file doesn't exist, create an empty, zero-byte file:

```
touch /etc/multipath.conf
```

2. The first time the `multipath.conf` file is created, you might need to enable and start the multipath services to load the recommended settings:

```
systemctl enable multipathd
```

```
systemctl start multipathd
```

3. Each time you boot the host, the empty `/etc/multipath.conf` zero-byte file automatically loads the NetApp recommended host multipath parameters as the default settings. You shouldn't need to make changes to the `/etc/multipath.conf` file for your host because the operating system is compiled with the multipath parameters that recognize and manage ONTAP LUNs correctly.

   The following table shows the Linux OS native compiled multipath parameter settings for ONTAP LUNs.

Show parameter settings

Parameter	Setting
detect_prio	yes
dev_loss_tmo	"infinity"
failback	immediate
fast_io_fail_tmo	5
features	"2 pg_init_retries 50"
flush_on_last_del	"yes"
hardware_handler	"0"
no_path_retry	queue
path_checker	"tur"
path_grouping_policy	"group_by_prio"
path_selector	"service-time 0"
polling_interval	5
prio	"ontap"
product	LUN
retain_attached_hw_handler	yes
rr_weight	"uniform"
user_friendly_names	no
vendor	NETAPP

4. Verify the parameter settings and path status for your ONTAP LUNs:

```
multipath -ll
```

The default multipath parameters support ASA, AFF, and FAS configurations. In these configurations, a single ONTAP LUN shouldn’t require more than four paths. Having more than four paths can cause problems during a storage failure.

The following example outputs show the correct parameter settings and path status for ONTAP LUNs in an ASA, AFF, or FAS configuration.

ASA configuration

An ASA configuration optimizes all paths to a given LUN, keeping them active. This improves performance by serving I/O operations through all paths at the same time.

Show example

```
multipath -ll
3600a098038314c4a433f577471797958 dm-2 NETAPP,LUN C-Mode
size=180G features='3 queue_if_no_path pg_init_retries 50'
hwhandler='1 alua' wp=rw
`-+- policy='service-time 0' prio=50 status=active
  |- 14:0:0:0  sdc  8:32   active ready running
  |- 17:0:0:0  sdas 66:192 active ready running
  |- 14:0:3:0  sdar 66:176 active ready running
  `- 17:0:3:0  sdch 69:80  active ready running
```

AFF or FAS configuration

An AFF or FAS configuration should have two groups of paths with higher and lower priorities. Higher priority Active/Optimized paths are served by the controller where the aggregate is located. Lower priority paths are active but non-optimized because they are served by a different controller. Non-optimized paths are only used when optimized paths aren't available.

The following example displays the output for an ONTAP LUN with two Active/Optimized paths and two Active/Non-Optimized paths:

Show example

```
multipath -ll
3600a0980383149764b5d567257516273 dm-0 NETAPP,LUN C-Mode
size=150G features='3 queue_if_no_path pg_init_retries 50'
hwhandler='1 alua' wp=rw
|-+- policy='service-time 0' prio=50 status=active
| |- 16:0:3:0  sdcg 69:64  active ready running
| `- 10:0:0:0  sdb  8:16   active ready running
`-+- policy='service-time 0' prio=10 status=enabled
  |- 10:0:1:0  sdc  8:32   active ready running
  `- 16:0:2:0  sdcf 69:48  active ready running
```

Step 4: Confirm the iSCSI configuration for your host

Ensure that iSCSI is configured correctly for your host.

About this task

You perform the following steps on the iSCSI host.

Steps

1. Verify that the iSCSI initiator package (iscsi-initiator-utils) is installed:

   ```
   rpm -qa | grep iscsi-initiator-utils
   ```

   You should see an output similar to the following example:

   ```
   iscsi-initiator-utils-6.2.1.11-0.git4b3e853.el9.x86_64
   ```

2. Verify the iSCSI initiator node name, which is located in the `/etc/iscsi/initiatorname.iscsi` file:

   ```
   InitiatorName=iqn.YYYY-MM.com.<vendor>:<host_name>
   ```

3. Configure the iSCSI session timeout parameter located in the `/etc/iscsi/iscsid.conf` file:

   ```
   node.session.timeo.replacement_timeout = 5
   ```

   The iSCSI `replacement_timeout` parameter controls how long the iSCSI layer should wait for a timed-out path or session to reestablish itself before failing any commands on it. You should set the value of `replacement_timeout` to 5 in the iSCSI configuration file.

4. Enable the iSCSI service:

   ```
   $systemctl enable iscsid
   ```

5. Start the iSCSI service:

   ```
   $systemctl start iscsid
   ```

6. Verify that the iSCSI service is running:

   ```
   $systemctl status iscsid
   ```

Show example

```
● iscsid.service - Open-iSCSI
     Loaded: loaded (/usr/lib/systemd/system/iscsid.service;
enabled; preset: disabled)
     Active: active (running) since Tue 2025-12-02 11:36:21 EST; 2
weeks 1 day ago
TriggeredBy: ● iscsid.socket
       Docs: man:iscsid(8)
             man:iscsiuio(8)
             man:iscsiadm(8)
   Main PID: 2263 (iscsid)
     Status: "Ready to process requests"
      Tasks: 1 (limit: 816061)
     Memory: 18.5M
        CPU: 14.480s
     CGroup: /system.slice/iscsid.service
             └─2263 /usr/sbin/iscsid -f -d2
```

7. Discover the iSCSI targets:

```
$iscsiadm --mode discovery --op update --type sendtargets --portal
<target_IP>
```

show example

```
iscsiadm --mode discovery --op update --type sendtargets --portal
192.168.30.87
192.168.30.87:3260,1139 iqn.1992-
08.com.netapp:sn.064a9b19b3ee11f09dcad039eabac370:vs.23
192.168.31.97:3260,1142 iqn.1992-
08.com.netapp:sn.064a9b19b3ee11f09dcad039eabac370:vs.23
192.168.31.87:3260,1141 iqn.1992-
08.com.netapp:sn.064a9b19b3ee11f09dcad039eabac370:vs.23
192.168.30.97:3260,1140 iqn.1992-
08.com.netapp:sn.064a9b19b3ee11f09dcad039eabac370:vs.23
```

8. Log in to the targets:

```
$iscsiadm --mode node -l all
```

9. Set iSCSI to log in automatically when the host boots:

```
$iscsiadm --mode node -T <target_name> -p <ip:port> -o update -n
node.startup -v automatic
```

You should see an output similar to the following example:

```
iscsiadm --mode node -T iqn.1992-
08.com.netapp:sn.064a9b19b3ee11f09dcad039eabac370:vs.23 -p
192.168.30.87:3260 -o update -n node.startup -v automatic
```

10. Verify the iSCSI sessions:

```
$iscsiadm --mode session
```

Show example

```
iscsiadm --mode session
tcp: [1] 192.168.30.87:3260,1139 iqn.1992-
08.com.netapp:sn.064a9b19b3ee11f09dcad039eabac370:vs.23 (non-flash)
tcp: [2] 192.168.31.97:3260,1142 iqn.1992-
08.com.netapp:sn.064a9b19b3ee11f09dcad039eabac370:vs.23 (non-flash)
tcp: [3] 192.168.31.87:3260,1141 iqn.1992-
08.com.netapp:sn.064a9b19b3ee11f09dcad039eabac370:vs.23 (non-flash)
tcp: [4] 192.168.30.97:3260,1140 iqn.1992-
08.com.netapp:sn.064a9b19b3ee11f09dcad039eabac370:vs.23 (non-flash)
```

Step 5: Optionally, exclude a device from multipathing

If required, you can exclude a device from multipathing by adding the WWID for the unwanted device to the "blacklist" stanza for the `multipath.conf` file.

Steps

1. Determine the WWID:

```
/lib/udev/scsi_id -gud /dev/sda
```

"sda" is the local SCSI disk that you want to add to the blacklist.

An example WWID is `360030057024d0730239134810c0cb833`.

2. Add the WWID to the "blacklist" stanza:

```
blacklist {
         wwid   360030057024d0730239134810c0cb833
        devnode "^(ram|raw|loop|fd|md|dm-|sr|scd|st)[0-9]*"
        devnode "^hd[a-z]"
        devnode "^cciss.*"
}
```

Step 6: Customize multipath parameters for ONTAP LUNs

If your host is connected to LUNs from other vendors and any of the multipath parameter settings are overridden, you need to correct them by adding stanzas later in the `multipath.conf` file that apply specifically to ONTAP LUNs. If you don't do this, the ONTAP LUNs might not work as expected.

Check your `/etc/multipath.conf` file, especially in the defaults section, for settings that might be overriding the default settings for multipath parameters.

> ⚠ You shouldn't override the recommended parameter settings for ONTAP LUNs. These settings are required for optimal performance of your host configuration. Contact NetApp support, your OS vendor, or both for more information.

The following example shows how to correct an overridden default. In this example, the `multipath.conf` file defines values for `path_checker` and `no_path_retry` that aren't compatible with ONTAP LUNs, and you can't remove these parameters because ONTAP storage arrays are still attached to the host. Instead, you correct the values for `path_checker` and `no_path_retry` by adding a device stanza to the `multipath.conf` file that applies specifically to the ONTAP LUNs.

Show example

```
defaults {
   path_checker       readsector0
   no_path_retry      fail
}

devices {
   device {
      vendor           "NETAPP"
      product          "LUN"
      no_path_retry    queue
      path_checker     tur
   }
}
```

Step 7: Review the known issues

RHEL 8.x with ONTAP storage has the following known issues.

8.1

NetApp Bug ID	Title	Description
1275843	Kernel disruption might occur on Red Hat Enterprise Linux 8.1 with QLogic QLE2672 16GB FC HBA during storage failover operation	Kernel disruption might occur during storage failover operations on the Red Hat Enterprise Linux 8.1 kernel with a QLogic QLE2672 Fibre Channel (FC) host bus adapter (HBA). The kernel disruption causes Red Hat Enterprise Linux 8.1 to reboot, leading to application disruption. If the kdump mechanism is enabled, the kernel disruption generates a vmcore file located in the/var/crash/ directory. You can check the vmcore file to determine the cause of the disruption.A storage failover with the QLogic QLE2672 HBA event affects the "kmem_cache_alloc+131" module. You can locate the event in the vmcore file by finding the following string: " [exception RIP: kmem_cache_alloc+131]"After the kernel disruption, reboot the Host OS and recover the operating system. Then restart the applications

NetApp Bug ID	Title	Description
1275838	Kernel disruption occurs on Red Hat Enterprise Linux 8.1 with QLogic QLE2742 32GB FC HBA during storage failover operations	Kernel disruption occurs during storage failover operations on the Red Hat Enterprise Linux 8.1 kernel with a QLogic QLE2742 Fibre Channel (FC) host bus adapter (HBA). The kernel disruption causes Red Hat Enterprise Linux 8.1 to reboot, leading to application disruption. If the kdump mechanism is enabled, the kernel disruption generates a vmcore file located in the/var/crash/ directory. You can check the vmcore file to determine the cause of the disruption. A storage failover with the QLogic QLE2742 HBA event affects the "kmem_cache_alloc+131"module. You can locate the event in the vmcore file by finding the following string: " [exception RIP: kmem_cache_alloc+131]"After the kernel disruption, reboot the Host OS and recover the operating system. Then restart the applications.
1266250	Login to multiple paths fails during the Red Hat Enterprise Linux 8.1 installation on iSCSI SAN LUN	You cannot login to multiple paths during the Red Hat Enterprise Linux 8.1 installation on iSCSI SAN LUN multipath devices. Installation is not possible on multipath iSCSI device and the multipath service is not enabled on the SAN boot device.

8.0

NetApp Bug ID	Title	Description
1238719	Kernel disruption on RHEL8 with QLogic QLE2672 16GB FC during storage failover operations	Kernel disruption might occur during storage failover operations on a Red Hat Enterprise Linux (RHEL) 8 kernel with a QLogic QLE2672 host bus adapter (HBA). The kernel disruption causes the operating system to reboot. The reboot causes application disruption and generates the vmcore file under the /var/crash/directory if kdump is configured. Use the vmcore file to identify the cause of the failure. In this case, the disruption is in the “kmem_cache_alloc+160” module. It is logged in the vmcore file with the following string: "[exception RIP: kmem_cache_alloc+160]". Reboot the host OS to recover the operating system and then restart the application.
1226783	RHEL8 OS boots up to "emergency mode" when more than 204 SCSI devices are mapped on all Fibre Channel (FC) host bus adapters (HBA)	If a host is mapped with more than 204 SCSI devices during an operating systemreboot process, the RHEL8 OS fails to boot up to "normal mode" and enters "emergency mode". This results in most of the host services becoming unavailable.
1230882	Creating a partition on an iSCSI multipath device during the RHEL8 installation is not feasible.	iSCSI SAN LUN multipath devices are not listed in disk selection during RHEL 8 installation. Consequently, the multipath service is not enabled on the SAN boot device.
1235998	The "rescan-scsi-bus.sh -a" command does not scan more than 328 devices	If a Red Hat Enterprise Linux 8 host maps with more than 328 SCSI devices, the host OS command "rescan-scsi-bus.sh -a" only scans 328 devices. The host does not discover any remaining mapped devices.

NetApp Bug ID	Title	Description
1231087	Remote ports transit to a blocked state on RHEL8 with Emulex LPe16002 16GB FC during storage failover operations	Remote ports transit to a blocked state on RHEL8 with Emulex LPe16002 16GB Fibre Channel (FC) during storage failover operations. When the storage node returns to an optimal state, the LIFs also come up and the remote port state should read "online". Occasionally, the remote port state might continue to read as "blocked" or "not present". This state can lead to a "failed faulty" path to LUNs at the multipath layer
1231098	Remote ports transit to blocked state on RHEL8 with Emulex LPe32002 32GB FC during storage failover operations	Remote ports transit to a blocked state on RHEL8 with Emulex LPe32002 32GBFibre Channel (FC) during storage failover operations. When the storage node returns to an optimal state, the LIFs also come up and the remote port state should read "online". Occasionally, the remote port state might continue to read as "blocked" or "not present". This state can lead to a "failed faulty" path to LUNs at the multipath layer.

What's next?

- Learn about using the Linux Host Utilities tool.
- Learn about ASM mirroring

  Automatic Storage Management (ASM) mirroring might require changes to the Linux multipath settings to allow ASM to recognize a problem and switch over to an alternate failure group. Most ASM configurations on ONTAP use external redundancy, which means that data protection is provided by the external array and ASM doesn't mirror data. Some sites use ASM with normal redundancy to provide two-way mirroring, normally across different sites. See Oracle databases on ONTAP for further information.

- Learn about Red Hat Linux Virtualization (KVM)

  Red Hat Linux can serve as a KVM host. This enables you to run multiple virtual machines on a single physical server using the Linux Kernel-based Virtual Machine (KVM) technology. The KVM host doesn't require explicit host configuration settings for ONTAP LUNs.

Rocky Linux

Configure Rocky Linux 10.x for FCP and iSCSI with ONTAP storage

The Linux Host Utilities software provides management and diagnostic tools for Linux hosts that are connected to ONTAP storage. When you install Linux Host Utilities on a Rocky Linux 10.x host, you can use the Host Utilities to help you manage FCP and iSCSI protocol operations with ONTAP LUNs.

Step 1: Optionally, enable SAN booting

You can configure your host to use SAN booting to simplify deployment and improve scalability.

Before you begin

Use the Interoperability Matrix Tool to verify that your Linux OS, host bus adapter (HBA), HBA firmware, HBA boot BIOS, and ONTAP version support SAN booting.

Steps

1. Create a SAN boot LUN and map it to the host.
2. Enable SAN booting in the server BIOS for the ports to which the SAN boot LUN is mapped.

   For information on how to enable the HBA BIOS, see your vendor-specific documentation.

3. Verify that the configuration was successful by rebooting the host and verifying that the OS is up and running.

Step 2: Install the Linux Host Utilities

NetApp strongly recommends installing the Linux Host Utilities to support ONTAP LUN management and assist technical support with gathering configuration data.

Install Linux Host Utilities 8.0.

Installing the Linux Host Utilities doesn't change any host timeout settings on your Linux host.

Step 3: Confirm the multipath configuration for your host

You can use multipathing with Rocky Linux 10.x to manage ONTAP LUNs.

To ensure that multipathing is configured correctly for your host, verify that the `/etc/multipath.conf` file is defined and that you have the NetApp recommended settings configured for your ONTAP LUNs.

Steps

1. Verify that the `/etc/multipath.conf` file exits. If the file doesn't exist, create an empty, zero-byte file:

   ```
   touch /etc/multipath.conf
   ```

2. The first time the `multipath.conf` file is created, you might need to enable and start the multipath services to load the recommended settings:

```
systemctl enable multipathd
```

```
systemctl start multipathd
```

3. Each time you boot the host, the empty `/etc/multipath.conf` zero-byte file automatically loads the NetApp recommended host multipath parameters as the default settings. You shouldn't need to make changes to the `/etc/multipath.conf` file for your host because the operating system is compiled with the multipath parameters that recognize and manage ONTAP LUNs correctly.

   The following table shows the Linux OS native compiled multipath parameter settings for ONTAP LUNs.

   Show parameter settings

Parameter	Setting
detect_prio	yes
dev_loss_tmo	"infinity"
failback	immediate
fast_io_fail_tmo	5
features	"2 pg_init_retries 50"
flush_on_last_del	"yes"
hardware_handler	"0"
no_path_retry	queue
path_checker	"tur"
path_grouping_policy	"group_by_prio"
path_selector	"service-time 0"
polling_interval	5
prio	"ontap"
product	LUN
retain_attached_hw_handler	yes
rr_weight	"uniform"
user_friendly_names	no
vendor	NETAPP

4. Verify the parameter settings and path status for your ONTAP LUNs:

```
multipath -ll
```

The default multipath parameters support ASA, AFF, and FAS configurations. In these configurations, a single ONTAP LUN shouldn’t require more than four paths. Having more than four paths can cause problems during a storage failure.

The following example outputs show the correct parameter settings and path status for ONTAP LUNs in an ASA, AFF, or FAS configuration.

ASA configuration

An ASA configuration optimizes all paths to a given LUN, keeping them active. This improves performance by serving I/O operations through all paths at the same time.

Show example

```
# multipath -ll
3600a098038314e535a24584e4b496252 dm-32 NETAPP,LUN C-Mode
size=10G features='3 queue_if_no_path pg_init_retries 50'
hwhandler='1 alua' wp=rw
`-+- policy='service-time 0' prio=50 status=active
  |- 11:0:0:41 sdan 66:112  active ready running
  |- 11:0:1:41 sdcb 68:240  active ready running
  |- 14:0:2:41 sdfd 129:240 active ready running
  `- 14:0:0:41 sddp 71:112  active ready running
```

AFF or FAS configuration

An AFF or FAS configuration should have two groups of paths with higher and lower priorities. Higher priority Active/Optimized paths are served by the controller where the aggregate is located. Lower priority paths are active but non-optimized because they are served by a different controller. Non-optimized paths are only used when optimized paths aren't available.

The following example displays the output for an ONTAP LUN with two Active/Optimized paths and two Active/Non-Optimized paths:

Show example

```
# multipath -ll
3600a0980383149764b5d567257516273 dm-0 NETAPP,LUN C-Mode
size=150G features='3 queue_if_no_path pg_init_retries 50'
hwhandler='1 alua' wp=rw
|-+- policy='service-time 0' prio=50 status=active
| |- 16:0:3:0  sdcg 69:64  active ready running
| `- 10:0:0:0  sdb  8:16   active ready running
`-+- policy='service-time 0' prio=10 status=enabled
  |- 10:0:1:0  sdc  8:32   active ready running
  `- 16:0:2:0  sdcf 69:48  active ready running
```

Step 4: Confirm the iSCSI configuration for your host

Ensure that iSCSI is configured correctly for your host.

About this task

You perform the following steps on the iSCSI host.

Steps

1. Verify that the iSCSI initiator package (iscsi-initiator-utils) is installed:

   ```
   rpm -qa | grep iscsi-initiator-utils
   ```

   You should see an output similar to the following example:

   ```
   iscsi-initiator-utils-6.2.1.11-0.git4b3e853.el9.x86_64
   ```

2. Verify the iSCSI initiator node name, which is located in the `/etc/iscsi/initiatorname.iscsi` file:

   ```
   InitiatorName=iqn.YYYY-MM.com.<vendor>:<host_name>
   ```

3. Configure the iSCSI session timeout parameter located in the `/etc/iscsi/iscsid.conf` file:

   ```
   node.session.timeo.replacement_timeout = 5
   ```

   The iSCSI `replacement_timeout` parameter controls how long the iSCSI layer should wait for a timed-out path or session to reestablish itself before failing any commands on it. You should set the value of `replacement_timeout` to 5 in the iSCSI configuration file.

4. Enable the iSCSI service:

   ```
   $systemctl enable iscsid
   ```

5. Start the iSCSI service:

   ```
   $systemctl start iscsid
   ```

6. Verify that the iSCSI service is running:

   ```
   $systemctl status iscsid
   ```

Show example

```
● iscsid.service - Open-iSCSI
     Loaded: loaded (/usr/lib/systemd/system/iscsid.service;
enabled; preset: disabled)
     Active: active (running) since Tue 2025-12-02 11:36:21 EST; 2
weeks 1 day ago
TriggeredBy: ● iscsid.socket
       Docs: man:iscsid(8)
             man:iscsiuio(8)
             man:iscsiadm(8)
   Main PID: 2263 (iscsid)
     Status: "Ready to process requests"
      Tasks: 1 (limit: 816061)
     Memory: 18.5M
        CPU: 14.480s
     CGroup: /system.slice/iscsid.service
             └─2263 /usr/sbin/iscsid -f -d2
```

7. Discover the iSCSI targets:

```
$iscsiadm --mode discovery --op update --type sendtargets --portal
<target_IP>
```

show example

```
iscsiadm --mode discovery --op update --type sendtargets --portal
192.168.30.87
192.168.30.87:3260,1139 iqn.1992-
08.com.netapp:sn.064a9b19b3ee11f09dcad039eabac370:vs.23
192.168.31.97:3260,1142 iqn.1992-
08.com.netapp:sn.064a9b19b3ee11f09dcad039eabac370:vs.23
192.168.31.87:3260,1141 iqn.1992-
08.com.netapp:sn.064a9b19b3ee11f09dcad039eabac370:vs.23
192.168.30.97:3260,1140 iqn.1992-
08.com.netapp:sn.064a9b19b3ee11f09dcad039eabac370:vs.23
```

8. Log in to the targets:

```
$iscsiadm --mode node -l all
```

9. Set iSCSI to log in automatically when the host boots:

```
$iscsiadm --mode node -T <target_name> -p <ip:port> -o update -n
node.startup -v automatic
```

You should see an output similar to the following example:

```
iscsiadm --mode node -T iqn.1992-
08.com.netapp:sn.064a9b19b3ee11f09dcad039eabac370:vs.23 -p
192.168.30.87:3260 -o update -n node.startup -v automatic
```

10. Verify the iSCSI sessions:

```
$iscsiadm --mode session
```

Show example

```
iscsiadm --mode session
tcp: [1] 192.168.30.87:3260,1139 iqn.1992-
08.com.netapp:sn.064a9b19b3ee11f09dcad039eabac370:vs.23 (non-flash)
tcp: [2] 192.168.31.97:3260,1142 iqn.1992-
08.com.netapp:sn.064a9b19b3ee11f09dcad039eabac370:vs.23 (non-flash)
tcp: [3] 192.168.31.87:3260,1141 iqn.1992-
08.com.netapp:sn.064a9b19b3ee11f09dcad039eabac370:vs.23 (non-flash)
tcp: [4] 192.168.30.97:3260,1140 iqn.1992-
08.com.netapp:sn.064a9b19b3ee11f09dcad039eabac370:vs.23 (non-flash)
```

Step 5: Optionally, exclude a device from multipathing

If required, you can exclude a device from multipathing by adding the WWID for the unwanted device to the "blacklist" stanza for the `multipath.conf` file.

Steps

1. Determine the WWID:

```
/lib/udev/scsi_id -gud /dev/sda
```

"sda" is the local SCSI disk that you want to add to the blacklist.

An example WWID is `360030057024d0730239134810c0cb833`.

2. Add the WWID to the "blacklist" stanza:

```
blacklist {
         wwid   360030057024d0730239134810c0cb833
        devnode "^(ram|raw|loop|fd|md|dm-|sr|scd|st)[0-9]*"
        devnode "^hd[a-z]"
        devnode "^cciss.*"
}
```

Step 6: Customize multipath parameters for ONTAP LUNs

If your host is connected to LUNs from other vendors and any of the multipath parameter settings are overridden, you need to correct them by adding stanzas later in the `multipath.conf` file that apply specifically to ONTAP LUNs. If you don't do this, the ONTAP LUNs might not work as expected.

Check your `/etc/multipath.conf` file, especially in the defaults section, for settings that might be overriding the default settings for multipath parameters.

> You shouldn't override the recommended parameter settings for ONTAP LUNs. These settings are required for optimal performance of your host configuration. Contact NetApp support, your OS vendor, or both for more information.

The following example shows how to correct an overridden default. In this example, the `multipath.conf` file defines values for `path_checker` and `no_path_retry` that aren't compatible with ONTAP LUNs, and you can't remove these parameters because ONTAP storage arrays are still attached to the host. Instead, you correct the values for `path_checker` and `no_path_retry` by adding a device stanza to the `multipath.conf` file that applies specifically to the ONTAP LUNs.

Show example

```
defaults {
   path_checker       readsector0
   no_path_retry      fail
}

devices {
   device {
      vendor           "NETAPP"
      product          "LUN"
      no_path_retry    queue
      path_checker     tur
   }
}
```

Step 7: Review the known issues

There are no known issues.

What's next?

- Learn about using the Linux Host Utilities tool.
- Learn about ASM mirroring

  Automatic Storage Management (ASM) mirroring might require changes to the Linux multipath settings to allow ASM to recognize a problem and switch over to an alternate failure group. Most ASM configurations on ONTAP use external redundancy, which means that data protection is provided by the external array and ASM doesn't mirror data. Some sites use ASM with normal redundancy to provide two-way mirroring, normally across different sites. See Oracle databases on ONTAP for further information.

- Learn about Rocky Linux Virtualization (KVM)

  Rocky Linux can serve as a KVM host. This enables you to run multiple virtual machines on a single physical server using the Linux Kernel-based Virtual Machine (KVM) technology. The KVM host doesn't require explicit host configuration settings for ONTAP LUNs.

Configure Rocky Linux 9.x for FCP and iSCSI with ONTAP storage

The Linux Host Utilities software provides management and diagnostic tools for Linux hosts that are connected to ONTAP storage. When you install Linux Host Utilities on a Rocky Linux 9.x host, you can use the Host Utilities to help you manage FCP and iSCSI protocol operations with ONTAP LUNs.

Step 1: Optionally, enable SAN booting

You can configure your host to use SAN booting to simplify deployment and improve scalability.

Before you begin

Use the Interoperability Matrix Tool to verify that your Linux OS, host bus adapter (HBA), HBA firmware, HBA boot BIOS, and ONTAP version support SAN booting.

Steps

1. Create a SAN boot LUN and map it to the host.
2. Enable SAN booting in the server BIOS for the ports to which the SAN boot LUN is mapped.

   For information on how to enable the HBA BIOS, see your vendor-specific documentation.

3. Verify that the configuration was successful by rebooting the host and verifying that the OS is up and running.

Step 2: Install the Linux Host Utilities

NetApp strongly recommends installing the Linux Host Utilities to support ONTAP LUN management and assist technical support with gathering configuration data.

Install Linux Host Utilities 8.0.

Installing the Linux Host Utilities doesn't change any host timeout settings on your Linux host.

Step 3: Confirm the multipath configuration for your host

You can use multipathing with Rocky Linux 9.x to manage ONTAP LUNs.

To ensure that multipathing is configured correctly for your host, verify that the `/etc/multipath.conf` file is defined and that you have the NetApp recommended settings configured for your ONTAP LUNs.

Steps

1. Verify that the `/etc/multipath.conf` file exits. If the file doesn't exist, create an empty, zero-byte file:

```
touch /etc/multipath.conf
```

2. The first time the `multipath.conf` file is created, you might need to enable and start the multipath services to load the recommended settings:

```
systemctl enable multipathd
```

```
systemctl start multipathd
```

3. Each time you boot the host, the empty `/etc/multipath.conf` zero-byte file automatically loads the NetApp recommended host multipath parameters as the default settings. You shouldn't need to make changes to the `/etc/multipath.conf` file for your host because the operating system is compiled with the multipath parameters that recognize and manage ONTAP LUNs correctly.

   The following table shows the Linux OS native compiled multipath parameter settings for ONTAP LUNs.

Show parameter settings

Parameter	Setting
detect_prio	yes
dev_loss_tmo	"infinity"
failback	immediate
fast_io_fail_tmo	5
features	"2 pg_init_retries 50"
flush_on_last_del	"yes"
hardware_handler	"0"
no_path_retry	queue
path_checker	"tur"
path_grouping_policy	"group_by_prio"
path_selector	"service-time 0"
polling_interval	5
prio	"ontap"
product	LUN
retain_attached_hw_handler	yes
rr_weight	"uniform"
user_friendly_names	no
vendor	NETAPP

4. Verify the parameter settings and path status for your ONTAP LUNs:

```
multipath -ll
```

The default multipath parameters support ASA, AFF, and FAS configurations. In these configurations, a single ONTAP LUN shouldn’t require more than four paths. Having more than four paths can cause problems during a storage failure.

The following example outputs show the correct parameter settings and path status for ONTAP LUNs in an ASA, AFF, or FAS configuration.

ASA configuration

An ASA configuration optimizes all paths to a given LUN, keeping them active. This improves performance by serving I/O operations through all paths at the same time.

Show example

```
multipath -ll
3600a098038314c4a433f577471797958 dm-2 NETAPP,LUN C-Mode
size=180G features='3 queue_if_no_path pg_init_retries 50'
hwhandler='1 alua' wp=rw
`-+- policy='service-time 0' prio=50 status=active
  |- 14:0:0:0  sdc  8:32   active ready running
  |- 17:0:0:0  sdas 66:192 active ready running
  |- 14:0:3:0  sdar 66:176 active ready running
  `- 17:0:3:0  sdch 69:80  active ready running
```

AFF or FAS configuration

An AFF or FAS configuration should have two groups of paths with higher and lower priorities. Higher priority Active/Optimized paths are served by the controller where the aggregate is located. Lower priority paths are active but non-optimized because they are served by a different controller. Non-optimized paths are only used when optimized paths aren't available.

The following example displays the output for an ONTAP LUN with two Active/Optimized paths and two Active/Non-Optimized paths:

Show example

```
multipath -ll
3600a0980383149764b5d567257516273 dm-0 NETAPP,LUN C-Mode
size=150G features='3 queue_if_no_path pg_init_retries 50'
hwhandler='1 alua' wp=rw
|-+- policy='service-time 0' prio=50 status=active
| |- 16:0:3:0  sdcg 69:64  active ready running
| `- 10:0:0:0  sdb  8:16   active ready running
`-+- policy='service-time 0' prio=10 status=enabled
  |- 10:0:1:0  sdc  8:32   active ready running
  `- 16:0:2:0  sdcf 69:48  active ready running
```

Step 4: Confirm the iSCSI configuration for your host

Ensure that iSCSI is configured correctly for your host.

About this task

You perform the following steps on the iSCSI host.

Steps

1. Verify that the iSCSI initiator package (iscsi-initiator-utils) is installed:

   ```
   rpm -qa | grep iscsi-initiator-utils
   ```

   You should see an output similar to the following example:

   ```
   iscsi-initiator-utils-6.2.1.11-0.git4b3e853.el9.x86_64
   ```

2. Verify the iSCSI initiator node name, which is located in the `/etc/iscsi/initiatorname.iscsi` file:

   ```
   InitiatorName=iqn.YYYY-MM.com.<vendor>:<host_name>
   ```

3. Configure the iSCSI session timeout parameter located in the `/etc/iscsi/iscsid.conf` file:

   ```
   node.session.timeo.replacement_timeout = 5
   ```

   The iSCSI `replacement_timeout` parameter controls how long the iSCSI layer should wait for a timed-out path or session to reestablish itself before failing any commands on it. You should set the value of `replacement_timeout` to 5 in the iSCSI configuration file.

4. Enable the iSCSI service:

   ```
   $systemctl enable iscsid
   ```

5. Start the iSCSI service:

   ```
   $systemctl start iscsid
   ```

6. Verify that the iSCSI service is running:

   ```
   $systemctl status iscsid
   ```

Show example

```
● iscsid.service - Open-iSCSI
     Loaded: loaded (/usr/lib/systemd/system/iscsid.service;
enabled; preset: disabled)
     Active: active (running) since Tue 2025-12-02 11:36:21 EST; 2
weeks 1 day ago
TriggeredBy: ● iscsid.socket
       Docs: man:iscsid(8)
             man:iscsiuio(8)
             man:iscsiadm(8)
   Main PID: 2263 (iscsid)
     Status: "Ready to process requests"
      Tasks: 1 (limit: 816061)
     Memory: 18.5M
        CPU: 14.480s
     CGroup: /system.slice/iscsid.service
             └─2263 /usr/sbin/iscsid -f -d2
```

7. Discover the iSCSI targets:

```
$iscsiadm --mode discovery --op update --type sendtargets --portal
<target_IP>
```

show example

```
iscsiadm --mode discovery --op update --type sendtargets --portal
192.168.30.87
192.168.30.87:3260,1139 iqn.1992-
08.com.netapp:sn.064a9b19b3ee11f09dcad039eabac370:vs.23
192.168.31.97:3260,1142 iqn.1992-
08.com.netapp:sn.064a9b19b3ee11f09dcad039eabac370:vs.23
192.168.31.87:3260,1141 iqn.1992-
08.com.netapp:sn.064a9b19b3ee11f09dcad039eabac370:vs.23
192.168.30.97:3260,1140 iqn.1992-
08.com.netapp:sn.064a9b19b3ee11f09dcad039eabac370:vs.23
```

8. Log in to the targets:

```
$iscsiadm --mode node -l all
```

9. Set iSCSI to log in automatically when the host boots:

```
$iscsiadm --mode node -T <target_name> -p <ip:port> -o update -n
node.startup -v automatic
```

You should see an output similar to the following example:

```
iscsiadm --mode node -T iqn.1992-
08.com.netapp:sn.064a9b19b3ee11f09dcad039eabac370:vs.23 -p
192.168.30.87:3260 -o update -n node.startup -v automatic
```

10. Verify the iSCSI sessions:

```
$iscsiadm --mode session
```

Show example

```
iscsiadm --mode session
tcp: [1] 192.168.30.87:3260,1139 iqn.1992-
08.com.netapp:sn.064a9b19b3ee11f09dcad039eabac370:vs.23 (non-flash)
tcp: [2] 192.168.31.97:3260,1142 iqn.1992-
08.com.netapp:sn.064a9b19b3ee11f09dcad039eabac370:vs.23 (non-flash)
tcp: [3] 192.168.31.87:3260,1141 iqn.1992-
08.com.netapp:sn.064a9b19b3ee11f09dcad039eabac370:vs.23 (non-flash)
tcp: [4] 192.168.30.97:3260,1140 iqn.1992-
08.com.netapp:sn.064a9b19b3ee11f09dcad039eabac370:vs.23 (non-flash)
```

Step 5: Optionally, exclude a device from multipathing

If required, you can exclude a device from multipathing by adding the WWID for the unwanted device to the "blacklist" stanza for the `multipath.conf` file.

Steps

1. Determine the WWID:

```
/lib/udev/scsi_id -gud /dev/sda
```

"sda" is the local SCSI disk that you want to add to the blacklist.

An example WWID is `360030057024d0730239134810c0cb833`.

2. Add the WWID to the "blacklist" stanza:

```
blacklist {
         wwid   360030057024d0730239134810c0cb833
        devnode "^(ram|raw|loop|fd|md|dm-|sr|scd|st)[0-9]*"
        devnode "^hd[a-z]"
        devnode "^cciss.*"
}
```

Step 6: Customize multipath parameters for ONTAP LUNs

If your host is connected to LUNs from other vendors and any of the multipath parameter settings are overridden, you need to correct them by adding stanzas later in the `multipath.conf` file that apply specifically to ONTAP LUNs. If you don't do this, the ONTAP LUNs might not work as expected.

Check your `/etc/multipath.conf` file, especially in the defaults section, for settings that might be overriding the default settings for multipath parameters.

> ⚠ You shouldn't override the recommended parameter settings for ONTAP LUNs. These settings are required for optimal performance of your host configuration. Contact NetApp support, your OS vendor, or both for more information.

The following example shows how to correct an overridden default. In this example, the `multipath.conf` file defines values for `path_checker` and `no_path_retry` that aren't compatible with ONTAP LUNs, and you can't remove these parameters because ONTAP storage arrays are still attached to the host. Instead, you correct the values for `path_checker` and `no_path_retry` by adding a device stanza to the `multipath.conf` file that applies specifically to the ONTAP LUNs.

Show example

```
defaults {
   path_checker       readsector0
   no_path_retry      fail
}

devices {
   device {
      vendor           "NETAPP"
      product          "LUN"
      no_path_retry    queue
      path_checker     tur
   }
}
```

Step 7: Review the known issues

Rocky Linux 9.x with ONTAP storage has the following known issues.

9.3

NetApp Bug ID	Title	Description	JIRA ID
1508554	NetApp Linux Host Utilities CLI requires additional library package dependencies to support Emulex host bus adapter (HBA) adapter discovery	In RHEL 9.x, the NetApp Linux SAN Host Utilities CLI `sanlun fcp show adapter -v` fails because the library package dependencies to support Emulex host bus adapter (HBA) discovery cannot be found.	Not applicable
1593771	A Red Hat Enterprise Linux 9.3 QLogic SAN host encounters loss of partial multipaths during storage mobility operations	During the ONTAP storage controller takeover operation, half of the multipaths are expected to go down or switch to a failover mode and then recover to full path count during the giveback workflow. However, with a Red Hat Enterprise Linux (RHEL) 9.3 QLogic host, only partial multipaths are recovered after a storage failover giveback operation.	RHEL 17811

9.2

NetApp Bug ID	Title	Description
1508554	NetApp Linux Host Utilities CLI requires additional library package dependencies to support Emulex HBA adapter discovery	In RHEL 9.2, the NetApp Linux SAN Host Utilities CLI `sanlun fcp show adapter -v` fails because the library package dependencies to support HBA discovery cannot be found.
1537359	A Red Hat Linux 9.2 SAN booted host with Emulex HBA encounters stalled tasks leading to kernel disruption	During a storage failover giveback operation, a Red Hat Linux 9.2 SAN booted host with an Emulex host bus adapeter (HBA) encounters stalled tasks leading to kernel disruption. The kernel disruption causes the operating system to reboot and if `kdump` is configured, it generates the `vmcore` file under the `/var/crash/` directory. The issue is being triaged with the `lpfc` driver but it cannot be reproduced consistently.

9.1

NetApp Bug ID	Title	Description
1508554	NetApp Linux Host Utilities CLI requires additional library package dependencies to support Emulex HBA adapter discovery	In Rocky Linux 9.1, the NetApp Linux SAN Host Utilities CLI `sanlun fcp show adapter -v` fails because the library package dependencies to support HBA discovery cannot be found.

What's next?

- Learn about using the Linux Host Utilities tool.
- Learn about ASM mirroring

  Automatic Storage Management (ASM) mirroring might require changes to the Linux multipath settings to allow ASM to recognize a problem and switch over to an alternate failure group. Most ASM configurations on ONTAP use external redundancy, which means that data protection is provided by the external array and ASM doesn't mirror data. Some sites use ASM with normal redundancy to provide two-way mirroring, normally across different sites. See Oracle databases on ONTAP for further information.

- Learn about Rocky Linux Virtualization (KVM)

  Rocky Linux can serve as a KVM host. This enables you to run multiple virtual machines on a single physical server using the Linux Kernel-based Virtual Machine (KVM) technology. The KVM host doesn't require explicit host configuration settings for ONTAP LUNs.

Configure Rocky Linux 8.x for FCP and iSCSI with ONTAP storage

The Linux Host Utilities software provides management and diagnostic tools for Linux hosts that are connected to ONTAP storage. When you install Linux Host Utilities on a Rocky Linux 8.x host, you can use the Host Utilities to help you manage FCP and iSCSI protocol operations with ONTAP LUNs.

Step 1: Optionally, enable SAN booting

You can configure your host to use SAN booting to simplify deployment and improve scalability.

Before you begin

Use the Interoperability Matrix Tool to verify that your Linux OS, host bus adapter (HBA), HBA firmware, HBA boot BIOS, and ONTAP version support SAN booting.

Steps

1. Create a SAN boot LUN and map it to the host.
2. Enable SAN booting in the server BIOS for the ports to which the SAN boot LUN is mapped.

   For information on how to enable the HBA BIOS, see your vendor-specific documentation.

3. Verify that the configuration was successful by rebooting the host and verifying that the OS is up and running.

Step 2: Install the Linux Host Utilities

NetApp strongly recommends installing the Linux Host Utilities to support ONTAP LUN management and assist technical support with gathering configuration data.

Install Linux Host Utilities 8.0.

Installing the Linux Host Utilities doesn't change any host timeout settings on your Linux host.

Step 3: Confirm the multipath configuration for your host

You can use multipathing with Rocky Linux 8.x to manage ONTAP LUNs.

To ensure that multipathing is configured correctly for your host, verify that the `/etc/multipath.conf` file is defined and that you have the NetApp recommended settings configured for your ONTAP LUNs.

Steps

1. Verify that the `/etc/multipath.conf` file exists:

   ```
   ls /etc/multipath.conf
   ```

   If the file doesn't exist, create an empty, zero-byte file:

   ```
   touch /etc/multipath.conf
   ```

2. The first time the `multipath.conf` file is created, you might need to enable and start the multipath services to load the recommended settings:

   ```
   systemctl enable multipathd
   ```

   ```
   systemctl start multipathd
   ```

3. Each time you boot the host, the empty `/etc/multipath.conf` zero-byte file automatically loads the NetApp recommended host multipath parameters as the default settings. You shouldn't need to make changes to the `/etc/multipath.conf` file for your host because the host operating system is compiled with the multipath parameters that recognize and manage ONTAP LUNs correctly.

   The following table shows the native Linux OS compiled multipath parameter settings for ONTAP LUNs.

Show parameter settings

Parameter	Setting
detect_prio	yes
dev_loss_tmo	"infinity"
failback	immediate
fast_io_fail_tmo	5
features	"2 pg_init_retries 50"
flush_on_last_del	"yes"
hardware_handler	"0"
no_path_retry	queue
path_checker	"tur"
path_grouping_policy	"group_by_prio"
path_selector	"service-time 0"
polling_interval	5
prio	"ontap"
product	LUN
retain_attached_hw_handler	yes
rr_weight	"uniform"
user_friendly_names	no
vendor	NETAPP

4. Verify the parameter settings and path status for your ONTAP LUNs:

```
multipath -ll
```

The default multipath parameters support ASA, AFF, and FAS configurations. In these configurations, a single ONTAP LUN shouldn’t require more than four paths. Having more than four paths can cause problems during a storage failure.

The following example outputs show the correct parameter settings and path status for ONTAP LUNs in an ASA, AFF, or FAS configuration.

ASA configuration

An ASA configuration optimizes all paths to a given LUN, keeping them active. This improves performance by serving I/O operations through all paths at the same time.

Show example

```
# multipath -ll
3600a098038314c4a433f577471797958 dm-2 NETAPP,LUN C-Mode
size=180G features='3 queue_if_no_path pg_init_retries 50'
hwhandler='1 alua' wp=rw
`-+- policy='service-time 0' prio=50 status=active
  |- 14:0:0:0  sdc  8:32   active ready running
  |- 17:0:0:0  sdas 66:192 active ready running
  |- 14:0:3:0  sdar 66:176 active ready running
  `- 17:0:3:0  sdch 69:80  active ready running
```

AFF or FAS configuration

An AFF or FAS configuration should have two groups of paths with higher and lower priorities. Higher priority Active/Optimized paths are served by the controller where the aggregate is located. Lower priority paths are active but non-optimized because they are served by a different controller. Non-optimized paths are only used when optimized paths aren't available.

The following example displays the output for an ONTAP LUN with two Active/Optimized paths and two Active/Non-Optimized paths:

Show example

```
# multipath -ll
3600a0980383149764b5d567257516273 dm-0 NETAPP,LUN C-Mode
size=150G features='3 queue_if_no_path pg_init_retries 50'
hwhandler='1 alua' wp=rw
|-+- policy='service-time 0' prio=50 status=active
| |- 16:0:3:0  sdcg 69:64  active ready running
| `- 10:0:0:0  sdb  8:16   active ready running
`-+- policy='service-time 0' prio=10 status=enabled
  |- 10:0:1:0  sdc  8:32   active ready running
  `- 16:0:2:0  sdcf 69:48  active ready running
```

Step 4: Confirm the iSCSI configuration for your host

Ensure that iSCSI is configured correctly for your host.

About this task

You perform the following steps on the iSCSI host.

Steps

1. Verify that the iSCSI initiator package (iscsi-initiator-utils) is installed:

   ```
   rpm -qa | grep iscsi-initiator-utils
   ```

   You should see an output similar to the following example:

   ```
   iscsi-initiator-utils-6.2.1.11-0.git4b3e853.el9.x86_64
   ```

2. Verify the iSCSI initiator node name, which is located in the `/etc/iscsi/initiatorname.iscsi` file:

   ```
   InitiatorName=iqn.YYYY-MM.com.<vendor>:<host_name>
   ```

3. Configure the iSCSI session timeout parameter located in the `/etc/iscsi/iscsid.conf` file:

   ```
   node.session.timeo.replacement_timeout = 5
   ```

   The iSCSI `replacement_timeout` parameter controls how long the iSCSI layer should wait for a timed-out path or session to reestablish itself before failing any commands on it. You should set the value of `replacement_timeout` to 5 in the iSCSI configuration file.

4. Enable the iSCSI service:

   ```
   $systemctl enable iscsid
   ```

5. Start the iSCSI service:

   ```
   $systemctl start iscsid
   ```

6. Verify that the iSCSI service is running:

   ```
   $systemctl status iscsid
   ```

Show example

```
● iscsid.service - Open-iSCSI
     Loaded: loaded (/usr/lib/systemd/system/iscsid.service;
enabled; preset: disabled)
     Active: active (running) since Tue 2025-12-02 11:36:21 EST; 2
weeks 1 day ago
TriggeredBy: ● iscsid.socket
       Docs: man:iscsid(8)
             man:iscsiuio(8)
             man:iscsiadm(8)
   Main PID: 2263 (iscsid)
     Status: "Ready to process requests"
      Tasks: 1 (limit: 816061)
     Memory: 18.5M
        CPU: 14.480s
     CGroup: /system.slice/iscsid.service
             └─2263 /usr/sbin/iscsid -f -d2
```

7. Discover the iSCSI targets:

```
$iscsiadm --mode discovery --op update --type sendtargets --portal
<target_IP>
```

show example

```
iscsiadm --mode discovery --op update --type sendtargets --portal
192.168.30.87
192.168.30.87:3260,1139 iqn.1992-
08.com.netapp:sn.064a9b19b3ee11f09dcad039eabac370:vs.23
192.168.31.97:3260,1142 iqn.1992-
08.com.netapp:sn.064a9b19b3ee11f09dcad039eabac370:vs.23
192.168.31.87:3260,1141 iqn.1992-
08.com.netapp:sn.064a9b19b3ee11f09dcad039eabac370:vs.23
192.168.30.97:3260,1140 iqn.1992-
08.com.netapp:sn.064a9b19b3ee11f09dcad039eabac370:vs.23
```

8. Log in to the targets:

```
$iscsiadm --mode node -l all
```

9. Set iSCSI to log in automatically when the host boots:

```
$iscsiadm --mode node -T <target_name> -p <ip:port> -o update -n
node.startup -v automatic
```

You should see an output similar to the following example:

```
iscsiadm --mode node -T iqn.1992-
08.com.netapp:sn.064a9b19b3ee11f09dcad039eabac370:vs.23 -p
192.168.30.87:3260 -o update -n node.startup -v automatic
```

10. Verify the iSCSI sessions:

```
$iscsiadm --mode session
```

Show example

```
iscsiadm --mode session
tcp: [1] 192.168.30.87:3260,1139 iqn.1992-
08.com.netapp:sn.064a9b19b3ee11f09dcad039eabac370:vs.23 (non-flash)
tcp: [2] 192.168.31.97:3260,1142 iqn.1992-
08.com.netapp:sn.064a9b19b3ee11f09dcad039eabac370:vs.23 (non-flash)
tcp: [3] 192.168.31.87:3260,1141 iqn.1992-
08.com.netapp:sn.064a9b19b3ee11f09dcad039eabac370:vs.23 (non-flash)
tcp: [4] 192.168.30.97:3260,1140 iqn.1992-
08.com.netapp:sn.064a9b19b3ee11f09dcad039eabac370:vs.23 (non-flash)
```

Step 5: Optionally, exclude a device from multipathing

If required, you can exclude a device from multipathing by adding the WWID for the unwanted device to the "blacklist" stanza for the `multipath.conf` file.

Steps

1. Determine the WWID:

```
/lib/udev/scsi_id -gud /dev/sda
```

"sda" is the local SCSI disk that you want to add to the blacklist.

An example WWID is `360030057024d0730239134810c0cb833`.

2. Add the WWID to the "blacklist" stanza:

```
blacklist {
         wwid   360030057024d0730239134810c0cb833
        devnode "^(ram|raw|loop|fd|md|dm-|sr|scd|st)[0-9]*"
        devnode "^hd[a-z]"
        devnode "^cciss.*"
}
```

Step 6: Customize multipath parameters for ONTAP LUNs

If your host is connected to LUNs from other vendors and any of the multipath parameter settings are overridden, you need to correct them by adding stanzas later in the `multipath.conf` file that apply specifically to ONTAP LUNs. If you don't do this, the ONTAP LUNs might not work as expected.

Check your `/etc/multipath.conf` file, especially in the defaults section, for settings that might be overriding the default settings for multipath parameters.

> You shouldn't override the recommended parameter settings for ONTAP LUNs. These settings are required for optimal performance of your host configuration. Contact NetApp support, your OS vendor, or both for more information.

The following example shows how to correct an overridden default. In this example, the `multipath.conf` file defines values for `path_checker` and `no_path_retry` that aren't compatible with ONTAP LUNs, and you can't remove these parameters because ONTAP storage arrays are still attached to the host. Instead, you correct the values for `path_checker` and `no_path_retry` by adding a device stanza to the `multipath.conf` file that applies specifically to the ONTAP LUNs.

Show example

```
defaults {
   path_checker       readsector0
   no_path_retry      fail
}

devices {
   device {
      vendor           "NETAPP"
      product          "LUN"
      no_path_retry    queue
      path_checker     tur
   }
}
```

Step 7: Review the known issues

There are no known issues.

What's next?

- Learn about using the Linux Host Utilities tool.
- Learn about ASM mirroring

  Automatic Storage Management (ASM) mirroring might require changes to the Linux multipath settings to allow ASM to recognize a problem and switch over to an alternate failure group. Most ASM configurations on ONTAP use external redundancy, which means that data protection is provided by the external array and ASM doesn't mirror data. Some sites use ASM with normal redundancy to provide two-way mirroring, normally across different sites. See Oracle databases on ONTAP for further information.

- Learn about Rocky Linux Virtualization (KVM)

  Rocky Linux can serve as a KVM host. This enables you to run multiple virtual machines on a single physical server using the Linux Kernel-based Virtual Machine (KVM) technology. The KVM host doesn't require explicit host configuration settings for ONTAP LUNs.

Solaris

Configure Solaris 11.4 for FCP and iSCSI with ONTAP storage

The Solaris Host Utilities software provides management and diagnostic tools for Solaris hosts that are connected to ONTAP storage. When you install the Solaris Host Utilities on a Solaris 11.4 host, you can use the Host Utilities to help you manage FCP and iSCSI protocol operations with ONTAP LUNs.

Step 1: Optionally, enable SAN booting

You can configure your host to use SAN booting to simplify deployment and improve scalability. If your configuration doesn't support SAN booting, you can use a local boot.

SAN boot

SAN booting is the process of setting up a SAN-attached disk (a LUN) as a boot device for a Solaris host. You can set up a SAN boot LUN to work in a Solaris MPxIO environment that is using the FC protocol and running Solaris Host Utilities. The method you use to set up a SAN boot LUN depends on your volume manager and file system.

Steps

1. Use the Interoperability Matrix Tool to verify that your Solaris OS, protocol, and ONTAP version support SAN booting.
2. Follow the best practices for setting up a SAN boot in the Solaris vendor documentation.

Local boot

Perform a local boot by installing the Solaris OS on the local hard disk, for example, install on an SSD, SATA, or RAID.

Step 2: Install the Solaris Host Utilities

NetApp strongly recommends installing the Solaris Host Utilities to support ONTAP LUN management and assist technical support with gathering configuration data.

Installing the Solaris Host Utilities changes some of the timeout settings on your Solaris host.

Install Solaris Host Utilities 8.0.

Step 3: Confirm the multipath configuration for your host

You can use multipathing with Solaris 11.4 to manage ONTAP LUNs.

Multipathing enables you to configure multiple network paths between the host and storage systems. If one path fails, traffic continues on the remaining paths. Oracle Solaris I/O Multipathing (MPxIO) is enabled by default for Solaris 11.4 and SPARC systems.

Steps

1. If your host is configured for FC, verify that the default setting in `/kernel/drv/fp.conf` is set to `mpxio-disable="no"`.
2. The Solaris Host Utilities load the NetApp recommended parameter settings for SPARC and x86_64 processors.

   Show parameter settings

Parameter	Value
throttle_max	8
not_ready_retries	300
busy_retries	30
reset_retries	30
throttle_min	2
timeout_retries	10
physical_block_size	4096
disksort	false
cache-nonvolatile	true

   For additional information about Solaris 11.4 system settings, see Oracle Support DOC ID: 2595926.1.

3. If your storage configuration includes MetroCluster, Oracle Solaris virtualization, or SnapMirror active sync, change the default settings:

MetroCluster

By default, the Solaris OS fails to execute the I/O operations after **20s** if all paths to a LUN are lost. This is controlled by the `fcp_offline_delay` parameter. The default value for `fcp_offline_delay` is the appropriate for standard ONTAP clusters. However, in MetroCluster configurations, you must increase the value of `fcp_offline_delay` to **120s** to ensure that I/O doesn't prematurely time out during operations, including unplanned failovers.

For additional information and recommended changes to default settings for MetroCluster, see the Knowledge Base article Solaris host support considerations in a MetroCluster configuration.

Oracle Solaris virtualization

- Solaris virtualization options include Solaris Logical Domains (also called LDOMs or Oracle VM Server for SPARC), Solaris Dynamic Domains, Solaris Zones, and Solaris Containers. These technologies are also referred to as "Oracle Virtual Machines".
- You can use multiple options together, for example, a Solaris Container within a particular Solaris Logical Domain.
- NetApp supports the use of Solaris virtualization technologies where the overall configuration is supported by Oracle and any partition with direct access to LUNs is listed on the IMT in a supported configuration. This includes root containers, LDOM I/O domains, and LDOM using NPIV to access LUNs.
- Partitions or virtual machines that use only virtualized storage resources, such as a `vdsk`, don't need specific qualifications because they don't have direct access to ONTAP LUNs. You only need to verify that the partition or virtual machine that has direct access to the underlying LUN, such as an LDOM I/O domain, is listed on the IMT.

Steps

When LUNs are used as virtual disk devices within an LDOM, the source of the LUN is masked by virtualization and the LDOM doesn't correctly detect the block sizes. To prevent this issue:

a. Patch the LDOM OS for *Oracle Bug 15824910*

b. Create a `vdc.conf` file that sets the block size of the virtual disk to `4096`. See Oracle DOC: 2157669.1 for more information.

c. Verify the patch installation to ensure that the recommended settings have been configured correctly:

   i. Create a zpool:

   ```
   zpool create zpool_name disk_list
   ```

   ii. Run `zdb -C` against the zpool and verify that the value of **ashift** is `12`.

   If the value of **ashift** isn't `12`, re run `zdb -C11`, and verify that the correct patch was installed and recheck the contents of `vdc.conf`.

   Don't proceed until **ashift** shows a value of `12`.

Patches are available for Oracle bug 15824910 on several Solaris versions. Contact Oracle if you need assistance in determining the best kernel patch.

SnapMirror active sync

Beginning with ONTAP 9.9.1, SnapMirror active sync setting configurations are supported in the Solaris host. To verify that the Solaris client applications are non-disruptive when an unplanned site failover switchover occurs in a SnapMirror active sync environment, you must configure the `scsi-vhci-failover-override` setting on the Solaris host. This setting overrides the failover module `f_tpgs` to prevent the execution of the code path that detects the contradiction.

Steps

a. Create the configuration file `/etc/driver/drv/scsi_vhci.conf` with an entry similar to the following example for the NetApp storage type connected to the host:

```
scsi-vhci-failover-override =
"NETAPP  LUN","f_tpgs"
```

b. Verify that the override parameter has been successfully applied:

```
devprop
```

```
mdb
```

Show examples

```
root@host-A:~# devprop -v -n /scsi_vhci scsi-vhci-failover-
override      scsi-vhci-failover-override=NETAPP  LUN + f_tpgs
root@host-A:~# echo "*scsi_vhci_dip::print -x struct dev_info
devi_child | ::list struct dev_info devi_sibling| ::print
struct dev_info devi_mdi_client| ::print mdi_client_t
ct_vprivate| ::print struct scsi_vhci_lun svl_lun_wwn
svl_fops_name"| mdb -k
```

```
svl_lun_wwn = 0xa002a1c8960 "600a0980383134775543f524539787938"
svl_fops_name = 0xa00298d69e0 "conf f_tpgs"
```

After `scsi-vhci-failover-override` has been applied, `conf` is added to `svl_fops_name`. For additional information and recommended changes to default settings, refer to the NetApp Knowledge Base article Solaris Host support recommended settings in SnapMirror active sync configuration.

4. Verify that 4KB aligned I/O with zpools using ONTAP LUNs is supported:

a. Verify that your Solaris host is installed with the latest Support Repository Update (SRU):

```
pkg info entire`
```

b. Verify that the ONTAP LUN has `ostype` as "Solaris", independent of the LUN size:

```
lun show -vserver` <vsersver_name>
```

Show example

```
chat-a800-31-33-35-37::*> lun show -vserver solaris_fcp -path
/vol/sol_195_zpool_vol_9/lun -fields ostype
vserver     path                          ostype
----------- ----------------------------- -------
solaris_fcp /vol/sol_195_zpool_vol_9/lun solaris
```

5. Verify the output for your ONTAP LUNs:

```
sanlun lun show
```

You should see an output similar to the following example for an ASA, AFF, or FAS configuration:

Show example

```
root@sparc-s7-55-148:~# sanlun lun show -pv

                    ONTAP Path: Solaris_148_siteA:/vol/Triage/lun
                           LUN: 0
                      LUN Size: 20g
                   Host Device:
/dev/rdsk/c0t600A098038314B32685D573064776172d0s2
                          Mode: C
              Multipath Provider: Sun Microsystems
                Multipath Policy: Native
```

6. Verify the path status for your ONTAP LUNs:

```
mpathadm show lu <LUN>`
```

The following example outputs show the correct path status for ONTAP LUNs in an ASA, AFF, or FAS configuration. The path priorities are displayed against "Access State" for each LUN in the output.

ASA configurations

An ASA configuration optimizes all paths to a given LUN, keeping them active. This improves performance by serving I/O operations through all paths at the same time.

Show example

```
root@sparc-s7-55-82:~# mpathadm show lu
/dev/rdsk/c0t600A098038313953495D58674777794Bd0s2
Logical Unit:  /dev/rdsk/c0t600A098038313953495D58674777794Bd0s2
        mpath-support:  libmpscsi_vhci.so
        Vendor:  NETAPP
        Product:  LUN C-Mode
        Revision:  9171
        Name Type:  unknown type
        Name:  600a098038313953495d58674777794b
        Asymmetric:  yes
        Current Load Balance:  round-robin
        Logical Unit Group ID:  NA
        Auto Failback:  on
        Auto Probing:  NA

        Paths:
                Initiator Port Name:  100000109bd30070
                Target Port Name:  20b9d039ea593393
                Logical Unit Number:  0
                Override Path:  NA
                Path State:  OK
                Disabled:  no

                Initiator Port Name:  100000109bd30070
                Target Port Name:  20b8d039ea593393
                Logical Unit Number:  0
                Override Path:  NA
                Path State:  OK
                Disabled:  no

                Initiator Port Name:  100000109bd3006f
                Target Port Name:  20b3d039ea593393
                Logical Unit Number:  0
                Override Path:  NA
                Path State:  OK
                Disabled:  no

                Initiator Port Name:  100000109bd3006f
                Target Port Name:  20b4d039ea593393
                Logical Unit Number:  0
                Override Path:  NA
                Path State:  OK
                Disabled:  no
```

```
        Target Port Groups:
                ID:  1003
                Explicit Failover:  no
                Access State:  active optimized
                Target Ports:
                        Name:  20b9d039ea593393
                        Relative ID:  8

                        Name:  20b4d039ea593393
                        Relative ID:  3

                ID:  1002
                Explicit Failover:  no
                Access State:  active optimized
                Target Ports:
                        Name:  20b8d039ea593393
                        Relative ID:  7

                        Name:  20b3d039ea593393
                        Relative ID:  2
```

AFF or FAS configuration

An AFF or FAS configuration should have two groups of paths with higher and lower priorities. Higher priority Active/Optimized paths are served by the controller where the aggregate is located. Lower priority paths are active but non-optimized because they are served by a different controller. Non-optimized paths are only used when optimized paths aren’t available.

The following example displays the correct output for an ONTAP LUN with two Active/Optimized paths and two Active/Non-Optimized paths:

Show example

```
root@chatsol-54-195:~# mpathadm show lu
/dev/rdsk/c0t600A0980383044376C3F4E694E506E44d0s2
Logical Unit:  /dev/rdsk/c0t600A0980383044376C3F4E694E506E44d0s2
        mpath-support:  libmpscsi_vhci.so
        Vendor:  NETAPP
        Product:  LUN C-Mode
        Revision:  9171
        Name Type:  unknown type
        Name:  600a0980383044376c3f4e694e506e44
        Asymmetric:  yes
        Current Load Balance:  round-robin
        Logical Unit Group ID:  NA
        Auto Failback:  on
        Auto Probing:  NA

        Paths:

                Initiator Port Name:  100000109b56c5fb
                Target Port Name:  205200a098ba7afe
                Logical Unit Number:  1
                Override Path:  NA
                Path State:  OK
                Disabled:  no

                Initiator Port Name:  100000109b56c5fb
                Target Port Name:  205000a098ba7afe
                Logical Unit Number:  1
                Override Path:  NA
                Path State:  OK
                Demoted:  yes
                Disabled:  no

                Initiator Port Name:  100000109b56c5fa
                Target Port Name:  204f00a098ba7afe
                Logical Unit Number:  1
                Override Path:  NA
                Path State:  OK
                Demoted:  yes
                Disabled:  no

                Initiator Port Name:  100000109b56c5fa
                Target Port Name:  205100a098ba7afe
                Logical Unit Number:  1
                Override Path:  NA
```

```
                Path State:  OK
                Disabled:  no

        Target Port Groups:
                ID:  1001
                Explicit Failover:  no
                Access State:  active not optimized
                Target Ports:
                        Name:  205200a098ba7afe
                        Relative ID:  8

                        Name:  205100a098ba7afe
                        Relative ID:  7

                ID:  1000
                Explicit Failover:  no
                Access State:  active optimized
                Target Ports:
                        Name:  205000a098ba7afe
                        Relative ID:  6

                        Name:  204f00a098ba7afe
                        Relative ID:  5
```

Step 4: Review the known issues

The Solaris 11.4 release for FCP and iSCSI with ONTAP storage has the following known issues:

NetApp Bug ID	Title	Description	Oracle ID
1362435	HUK 6.2 and Solaris_11.4 FC driver binding changes	Refer to Solaris 11.4 and HUK recommendations. FC driver binding is changed from `ssd (4D)` to `sd (4D)`. Move the existing configuration from `ssd.conf` to `sd.conf` as mentioned in Oracle DOC: 2595926.1). The behavior varies across newly installed Solaris 11.4 systems and systems upgraded from Solaris 11.3 or earlier versions.	(Doc ID 2595926.1)

NetApp Bug ID	Title	Description	Oracle ID
1366780	Solaris LIF issue noticed during storage failover (SFO) giveback operation with Emulex 32G host bus adapter (HBA) on x86 Arch	Solaris LIF issue noticed with Emulex firmware version 12.6.x and later on the x86_64 platform.	SR 3-24746803021
1368957	Solaris 11.x `cfgadm -c configure` resulting in I/O error with end-to-end Emulex configuration	Running `cfgadm -c configure` on Emulex end-to-end configuration results in an I/O error. This is fixed in ONTAP 9.5P17, 9.6P14 , 9.7P13, and 9.8P2	Not Applicable
1345622	Abnormal path reporting on Solaris hosts with ASA/PPorts using OS native commands	Intermittent path reporting issues are noticed on Solaris 11.4 with All SAN Array (ASA).	Not Applicable

What's next?

Learn about using the Solaris Host Utilities tool.

Configure Solaris 11.3 for FCP and iSCSI with ONTAP storage

The Solaris Host Utilities software provides management and diagnostic tools for Solaris hosts that are connected to ONTAP storage. When you install the Solaris Host Utilities on a Solaris 11.3 host, you can use the Host Utilities to help you manage FCP and iSCSI protocol operations with ONTAP LUNs.

Step 1: Optionally, enable SAN booting

You can configure your host to use SAN booting to simplify deployment and improve scalability. If your configuration doesn't support SAN booting, you can use a local boot.

SAN boot

SAN booting is the process of setting up a SAN-attached disk (a LUN) as a boot device for a Solaris host. You can set up a SAN boot LUN to work in a Solaris MPxIO environment that is using the FC protocol and running Solaris Host Utilities. The method you use to set up a SAN boot LUN depends on your volume manager and file system.

Steps

1. Use the Interoperability Matrix Tool to verify that your Solaris OS, protocol, and ONTAP version support SAN booting.
2. Follow the best practices for setting up a SAN boot in the Solaris vendor documentation.

Local boot

Perform a local boot by installing the Solaris OS on the local hard disk, for example, install on an SSD, SATA, or RAID.

Step 2: Install the Solaris Host Utilities

NetApp strongly recommends installing the Solaris Host Utilities to support ONTAP LUN management and assist technical support with gathering configuration data.

Installing the Solaris Host Utilities changes some of the timeout settings on your Solaris host.

Install Solaris Host Utilities 6.2.

Step 3: Confirm the multipath configuration for your host

You can use multipathing with Solaris 11.3 to manage ONTAP LUNs.

Multipathing enables you to configure multiple network paths between the host and storage systems. If one path fails, traffic continues on the remaining paths.

Steps

1. The Solaris Host Utilities load the NetApp recommended parameter settings for SPARC and x86_64 processors.

Show parameter settings

Parameter	Value
throttle_max	8
not_ready_retries	300
busy_retries	30
reset_retries	30
throttle_min	2
timeout_retries	10
physical_block_size	4096
disksort	false
cache-nonvolatile	true

2. If your storage configuration includes MetroCluster, Oracle Solaris virtualization, or SnapMirror active sync, change the default settings:

MetroCluster

By default, the Solaris OS fails to execute the I/O operations after **20s** if all paths to a LUN are lost. This is controlled by the `fcp_offline_delay` parameter. The default value for `fcp_offline_delay` is the appropriate for standard ONTAP clusters. However, in MetroCluster configurations, you must increase the value of `fcp_offline_delay` to **120s** to ensure that I/O doesn't prematurely time out during operations, including unplanned failovers.

For additional information and recommended changes to default settings for MetroCluster, see the Knowledge Base article Solaris host support considerations in a MetroCluster configuration.

Oracle Solaris virtualization

- Solaris virtualization options include Solaris Logical Domains (also called LDOMs or Oracle VM Server for SPARC), Solaris Dynamic Domains, Solaris Zones, and Solaris Containers. These technologies are also referred to as "Oracle Virtual Machines".
- You can use multiple options together, for example, a Solaris Container within a particular Solaris Logical Domain.
- NetApp supports the use of Solaris virtualization technologies where the overall configuration is supported by Oracle and any partition with direct access to LUNs is listed on the IMT in a supported configuration. This includes root containers, LDOM I/O domains, and LDOM using NPIV to access LUNs.
- Partitions or virtual machines that use only virtualized storage resources, such as a `vdsk`, don't need specific qualifications because they don't have direct access to ONTAP LUNs. You only need to verify that the partition or virtual machine that has direct access to the underlying LUN, such as an LDOM I/O domain, is listed on the IMT.

Steps

When LUNs are used as virtual disk devices within an LDOM, the source of the LUN is masked by virtualization and the LDOM doesn't correctly detect the block sizes. To prevent this issue:

a. Patch the LDOM OS for *Oracle Bug 15824910*

b. Create a `vdc.conf` file that sets the block size of the virtual disk to `4096`. See Oracle DOC: 2157669.1 for more information.

c. Verify the patch installation to ensure that the recommended settings have been configured correctly:

   i. Create a zpool:

   ```
   zpool create zpool_name disk_list
   ```

   ii. Run `zdb -C` against the zpool and verify that the value of **ashift** is `12`.

   If the value of **ashift** isn't `12`, re run `zdb -C11`, and verify that the correct patch was installed and recheck the contents of `vdc.conf`.

   Don't proceed until **ashift** shows a value of `12`.

Patches are available for Oracle bug 15824910 on several Solaris versions. Contact Oracle if you need assistance in determining the best kernel patch.

SnapMirror active sync

Beginning with ONTAP 9.9.1, SnapMirror active sync setting configurations are supported in the Solaris host. To verify that the Solaris client applications are non-disruptive when an unplanned site failover switchover occurs in a SnapMirror active sync environment, you must configure the `scsi-vhci-failover-override` setting on the Solaris host. This setting overrides the failover module `f_tpgs` to prevent the execution of the code path that detects the contradiction.

Steps

a. Create the configuration file `/etc/driver/drv/scsi_vhci.conf` with an entry similar to the following example for the NetApp storage type connected to the host:

```
scsi-vhci-failover-override =
"NETAPP  LUN","f_tpgs"
```

b. Verify that the override parameter has been successfully applied:

```
devprop
```

```
mdb
```

Show examples

```
root@host-A:~# devprop -v -n /scsi_vhci scsi-vhci-failover-
override      scsi-vhci-failover-override=NETAPP  LUN + f_tpgs
root@host-A:~# echo "*scsi_vhci_dip::print -x struct dev_info
devi_child | ::list struct dev_info devi_sibling| ::print
struct dev_info devi_mdi_client| ::print mdi_client_t
ct_vprivate| ::print struct scsi_vhci_lun svl_lun_wwn
svl_fops_name"| mdb -k
```

```
svl_lun_wwn = 0xa002a1c8960 "600a0980383134775543f524539787938"
svl_fops_name = 0xa00298d69e0 "conf f_tpgs"
```

After `scsi-vhci-failover-override` has been applied, `conf` is added to `svl_fops_name`. For additional information and recommended changes to default settings, refer to the NetApp Knowledge Base article Solaris Host support recommended settings in SnapMirror active sync configuration.

3. Verify that 4KB aligned I/O with zpools using ONTAP LUNs is supported:

a. Verify that your Solaris host is installed with the latest Support Repository Update (SRU):

```
pkg info entire`
```

b. Verify that the ONTAP LUN has `ostype` as "Solaris", independent of the LUN size:

```
lun show -vserver` <vsersver_name>
```

Show example

```
chat-a800-31-33-35-37::*> lun show -vserver solaris_fcp -path
/vol/sol_195_zpool_vol_9/lun -fields ostype
vserver     path                          ostype
----------- ----------------------------- -------
solaris_fcp /vol/sol_195_zpool_vol_9/lun solaris
```

4. Verify the output for your ONTAP LUNs:

```
sanlun lun show
```

You should see an output similar to the following example for an ASA, AFF, or FAS configuration:

Show example

```
root@sparc-s7-55-148:~# sanlun lun show -pv

                    ONTAP Path: Solaris_148_siteA:/vol/Triage/lun
                           LUN: 0
                      LUN Size: 20g
                   Host Device:
/dev/rdsk/c0t600A098038314B32685D573064776172d0s2
                          Mode: C
              Multipath Provider: Sun Microsystems
                Multipath Policy: Native
```

5. Verify the path status for your ONTAP LUNs:

```
mpathadm show lu <LUN>`
```

The following example outputs show the correct path status for ONTAP LUNs in an ASA, AFF, or FAS configuration. The path priorities are displayed against "Access State" for each LUN in the output.

ASA configurations

An ASA configuration optimizes all paths to a given LUN, keeping them active. This improves performance by serving I/O operations through all paths at the same time.

Show example

```
root@sparc-s7-55-82:~# mpathadm show lu
/dev/rdsk/c0t600A098038313953495D58674777794Bd0s2
Logical Unit:  /dev/rdsk/c0t600A098038313953495D58674777794Bd0s2
        mpath-support:  libmpscsi_vhci.so
        Vendor:  NETAPP
        Product:  LUN C-Mode
        Revision:  9171
        Name Type:  unknown type
        Name:  600a098038313953495d58674777794b
        Asymmetric:  yes
        Current Load Balance:  round-robin
        Logical Unit Group ID:  NA
        Auto Failback:  on
        Auto Probing:  NA

        Paths:
                Initiator Port Name:  100000109bd30070
                Target Port Name:  20b9d039ea593393
                Logical Unit Number:  0
                Override Path:  NA
                Path State:  OK
                Disabled:  no

                Initiator Port Name:  100000109bd30070
                Target Port Name:  20b8d039ea593393
                Logical Unit Number:  0
                Override Path:  NA
                Path State:  OK
                Disabled:  no

                Initiator Port Name:  100000109bd3006f
                Target Port Name:  20b3d039ea593393
                Logical Unit Number:  0
                Override Path:  NA
                Path State:  OK
                Disabled:  no

                Initiator Port Name:  100000109bd3006f
                Target Port Name:  20b4d039ea593393
                Logical Unit Number:  0
                Override Path:  NA
                Path State:  OK
                Disabled:  no
```

```
        Target Port Groups:
                ID:  1003
                Explicit Failover:  no
                Access State:  active optimized
                Target Ports:
                        Name:  20b9d039ea593393
                        Relative ID:  8

                        Name:  20b4d039ea593393
                        Relative ID:  3

                ID:  1002
                Explicit Failover:  no
                Access State:  active optimized
                Target Ports:
                        Name:  20b8d039ea593393
                        Relative ID:  7

                        Name:  20b3d039ea593393
                        Relative ID:  2
```

AFF or FAS configuration

An AFF or FAS configuration should have two groups of paths with higher and lower priorities. Higher priority Active/Optimized paths are served by the controller where the aggregate is located. Lower priority paths are active but non-optimized because they are served by a different controller. Non-optimized paths are only used when optimized paths aren’t available.

The following example displays the correct output for an ONTAP LUN with two Active/Optimized paths and two Active/Non-Optimized paths:

Show example

```
root@chatsol-54-195:~# mpathadm show lu
/dev/rdsk/c0t600A0980383044376C3F4E694E506E44d0s2
Logical Unit:  /dev/rdsk/c0t600A0980383044376C3F4E694E506E44d0s2
        mpath-support:  libmpscsi_vhci.so
        Vendor:  NETAPP
        Product:  LUN C-Mode
        Revision:  9171
        Name Type:  unknown type
        Name:  600a0980383044376c3f4e694e506e44
        Asymmetric:  yes
        Current Load Balance:  round-robin
        Logical Unit Group ID:  NA
        Auto Failback:  on
        Auto Probing:  NA

        Paths:

                Initiator Port Name:  100000109b56c5fb
                Target Port Name:  205200a098ba7afe
                Logical Unit Number:  1
                Override Path:  NA
                Path State:  OK
                Disabled:  no

                Initiator Port Name:  100000109b56c5fb
                Target Port Name:  205000a098ba7afe
                Logical Unit Number:  1
                Override Path:  NA
                Path State:  OK
                Demoted:  yes
                Disabled:  no

                Initiator Port Name:  100000109b56c5fa
                Target Port Name:  204f00a098ba7afe
                Logical Unit Number:  1
                Override Path:  NA
                Path State:  OK
                Demoted:  yes
                Disabled:  no

                Initiator Port Name:  100000109b56c5fa
                Target Port Name:  205100a098ba7afe
                Logical Unit Number:  1
                Override Path:  NA
```

```
                Path State:  OK
                Disabled:  no

        Target Port Groups:
                ID:  1001
                Explicit Failover:  no
                Access State:  active not optimized
                Target Ports:
                        Name:  205200a098ba7afe
                        Relative ID:  8

                        Name:  205100a098ba7afe
                        Relative ID:  7

                ID:  1000
                Explicit Failover:  no
                Access State:  active optimized
                Target Ports:
                        Name:  205000a098ba7afe
                        Relative ID:  6

                        Name:  204f00a098ba7afe
                        Relative ID:  5
```

Step 4: Review the known issues

The Solaris 11.3 release for FCP and iSCSI with ONTAP storage has the following known issues:

NetApp Bug ID	Title	Description	Oracle ID
1366780	Solaris LIF problem during GB with Emulex 32G HBA on x86 Arch	Seen with Emulex Firmware version 12.6.x and later on x86_64 Platform	SR 3-24746803021
1368957	Solaris 11.x 'cfgadm -c configure' resulting in I/O error with End-to-End Emulex configuration	Running `cfgadm -c configure` on Emulex end-to-end configurations results in I/O error. This is fixed in ONTAP 9.5P17, 9.6P14, 9.7P13 and 9.8P2	Not Applicable

What's next?

Learn about using the Solaris Host Utilities tool.

SUSE Linux Enterprise Server

Configure SUSE Linux Enterprise Server 16 for FCP and iSCSI with ONTAP storage

The Linux Host Utilities software provides management and diagnostic tools for Linux hosts that are connected to ONTAP storage. When you install the Linux Host Utilities on a SUSE Linux Enterprise Server 16 host, you can use the Host Utilities to help you manage FCP and iSCSI protocol operations with ONTAP LUNs.

Step 1: Optionally, enable SAN booting

You can configure your host to use SAN booting to simplify deployment and improve scalability.

Before you begin

Use the Interoperability Matrix Tool to verify that your Linux OS, host bus adapter (HBA), HBA firmware, HBA boot BIOS, and ONTAP version support SAN booting.

Steps

1. Create a SAN boot LUN and map it to the host.
2. Enable SAN booting in the server BIOS for the ports to which the SAN boot LUN is mapped.

   For information on how to enable the HBA BIOS, see your vendor-specific documentation.
3. Verify that the configuration was successful by rebooting the host and verifying that the OS is up and running.

Step 2: Install the Linux Host Utilities

NetApp strongly recommends installing the Linux Host Utilities to support ONTAP LUN management and assist technical support with gathering configuration data.

Install Linux Host Utilities 7.1.

Installing the Linux Host Utilities doesn't change any host timeout settings on your Linux host.

Step 3: Confirm the multipath configuration for your host

You can use multipathing with SUSE Linux Enterprise Server 16 to manage ONTAP LUNs.

To ensure that multipathing is configured correctly for your host, verify that the `/etc/multipath.conf` file is defined and that you have the NetApp recommended settings configured for your ONTAP LUNs.

Steps

1. Verify that the `/etc/multipath.conf` file exists:

   ```
   ls /etc/multipath.conf
   ```

   If the file doesn't exist, create an empty, zero-byte file:

```
touch /etc/multipath.conf
```

2. The first time the `multipath.conf` file is created, you might need to enable and start the multipath services to load the recommended settings:

```
systemctl enable multipathd
```

```
systemctl start multipathd
```

3. Each time you boot the host, the empty `/etc/multipath.conf` zero-byte file automatically loads the NetApp recommended host multipath parameters as the default settings. You shouldn't need to make changes to the `/etc/multipath.conf` file for your host because the host operating system is compiled with the multipath parameters that recognize and manage ONTAP LUNs correctly.

   The following table shows the native Linux OS compiled multipath parameter settings for ONTAP LUNs.

   Show parameter settings

Parameter	Setting
detect_prio	yes
dev_loss_tmo	"infinity"
failback	immediate
fast_io_fail_tmo	5
features	"2 pg_init_retries 50"
flush_on_last_del	"yes"
hardware_handler	"0"
no_path_retry	queue
path_checker	"tur"
path_grouping_policy	"group_by_prio"
path_selector	"service-time 0"
polling_interval	5
prio	"ontap"
product	LUN
retain_attached_hw_handler	yes
rr_weight	"uniform"
user_friendly_names	no
vendor	NETAPP

4. Verify the parameter settings and path status for your ONTAP LUNs:

```
multipath -ll
```

The default multipath parameters support ASA, AFF, and FAS configurations. In these configurations, a single ONTAP LUN shouldn’t require more than four paths. Having more than four paths can cause problems during a storage failure.

The following example outputs show the correct parameter settings and path status for ONTAP LUNs in an ASA, AFF, or FAS configuration.

ASA configuration

An ASA configuration optimizes all paths to a given LUN, keeping them active. This improves performance by serving I/O operations through all paths at the same time.

Show example

```
# multipath -ll
3600a09803831347657244e527766394e dm-5 NETAPP,LUN C-Mode
size=80G features='3 queue_if_no_path pg_init_retries 50'
hwhandler='1 alua' wp=rw
`-+- policy='service-time 0' prio=50 status=active
  |- 3:0:7:9    sdco 69:192  active ready running
  |- 3:0:8:9    sddi 71:0    active ready running
  |- 14:0:8:9   sdjq 65:320  active ready running
  `- 14:0:7:9   sdiw 8:256   active ready running
```

AFF or FAS configuration

An AFF or FAS configuration should have two groups of paths with higher and lower priorities. Higher priority Active/Optimized paths are served by the controller where the aggregate is located. Lower priority paths are active but non-optimized because they are served by a different controller. Non-optimized paths are only used when optimized paths aren’t available.

The following example displays the output for an ONTAP LUN with two Active/Optimized paths and two Active/Non-Optimized paths:

Show example

```
# multipath -ll
3600a09803831347657244e527766394e dm-5 NETAPP,LUN C-Mode
size=80G features='3 queue_if_no_path pg_init_retries 50'
hwhandler='1 alua' wp=rw
|-+- policy='service-time 0' prio=50 status=active
| |- 3:0:3:0    sdd  8:48    active ready running
| |- 3:0:4:0    sdx  65:112  active ready running
`-+- policy='service-time 0' prio=10 status=enabled
  |- 14:0:2:0   sdfk 130:96  active ready running
  `- 14:0:5:0   sdgz 132:240 active ready running
```

Step 4: Confirm the iSCSI configuration for your host

Ensure that iSCSI is configured correctly for your host.

About this task

You perform the following steps on the iSCSI host.

Steps

1. Verify that the iSCSI initiator package (iscsi-initiator-utils) is installed:

   ```
   rpm -qa | grep open-iscsi
   ```

   You should see an output similar to the following example:

   ```
   open-iscsi-2.1.11-160000.2.2.x86_64
   ```

2. Verify the iSCSI initiator node name, which is located in the `/etc/iscsi/initiatorname.iscsi` file:

   ```
   InitiatorName=iqn.YYYY-MM.com.<vendor>:<host_name>
   ```

3. Configure the iSCSI session timeout parameter located in the `/etc/iscsi/iscsid.conf` file:

   ```
   node.session.timeo.replacement_timeout = 5
   ```

   The iSCSI `replacement_timeout` parameter controls how long the iSCSI layer should wait for a timed-out path or session to reestablish itself before failing any commands on it. You should set the value of `replacement_timeout` to 5 in the iSCSI configuration file.

4. Enable the iSCSI service:

   ```
   $systemctl enable iscsid
   ```

5. Start the iSCSI service:

   ```
   $systemctl start iscsid
   ```

6. Verify that the iSCSI service is running:

   ```
   $systemctl status iscsid
   ```

Show example

```
● iscsid.service - Open-iSCSI
     Loaded: loaded (/usr/lib/systemd/system/iscsid.service;
enabled; preset: disabled)
     Active: active (running) since Tue 2025-12-02 11:36:21 EST; 2
weeks 1 day ago
TriggeredBy: ● iscsid.socket
       Docs: man:iscsid(8)
             man:iscsiuio(8)
             man:iscsiadm(8)
   Main PID: 2263 (iscsid)
     Status: "Ready to process requests"
      Tasks: 1 (limit: 816061)
     Memory: 18.5M
        CPU: 14.480s
     CGroup: /system.slice/iscsid.service
             └─2263 /usr/sbin/iscsid -f -d2
```

7. Discover the iSCSI targets:

```
$iscsiadm --mode discovery --op update --type sendtargets --portal
<target_IP>
```

show example

```
iscsiadm --mode discovery --op update --type sendtargets --portal
192.168.30.87
192.168.30.87:3260,1139 iqn.1992-
08.com.netapp:sn.064a9b19b3ee11f09dcad039eabac370:vs.23
192.168.31.97:3260,1142 iqn.1992-
08.com.netapp:sn.064a9b19b3ee11f09dcad039eabac370:vs.23
192.168.31.87:3260,1141 iqn.1992-
08.com.netapp:sn.064a9b19b3ee11f09dcad039eabac370:vs.23
192.168.30.97:3260,1140 iqn.1992-
08.com.netapp:sn.064a9b19b3ee11f09dcad039eabac370:vs.23
```

8. Log in to the targets:

```
$iscsiadm --mode node -l all
```

9. Set iSCSI to log in automatically when the host boots:

```
$iscsiadm --mode node -T <target_name> -p <ip:port> -o update -n
node.startup -v automatic
```

You should see an output similar to the following example:

```
iscsiadm --mode node -T iqn.1992-
08.com.netapp:sn.064a9b19b3ee11f09dcad039eabac370:vs.23 -p
192.168.30.87:3260 -o update -n node.startup -v automatic
```

10. Verify the iSCSI sessions:

```
$iscsiadm --mode session
```

Show example

```
iscsiadm --mode session
tcp: [1] 192.168.30.87:3260,1139 iqn.1992-
08.com.netapp:sn.064a9b19b3ee11f09dcad039eabac370:vs.23 (non-flash)
tcp: [2] 192.168.31.97:3260,1142 iqn.1992-
08.com.netapp:sn.064a9b19b3ee11f09dcad039eabac370:vs.23 (non-flash)
tcp: [3] 192.168.31.87:3260,1141 iqn.1992-
08.com.netapp:sn.064a9b19b3ee11f09dcad039eabac370:vs.23 (non-flash)
tcp: [4] 192.168.30.97:3260,1140 iqn.1992-
08.com.netapp:sn.064a9b19b3ee11f09dcad039eabac370:vs.23 (non-flash)
```

Step 5: Optionally, exclude a device from multipathing

If required, you can exclude a device from multipathing by adding the WWID for the unwanted device to the "blacklist" stanza for the `multipath.conf` file.

Steps

1. Determine the WWID:

```
/lib/udev/scsi_id -gud /dev/sda
```

"sda" is the local SCSI disk that you want to add to the blacklist.

An example WWID is `360030057024d0730239134810c0cb833`.

2. Add the WWID to the "blacklist" stanza:

```
blacklist {
         wwid   360030057024d0730239134810c0cb833
        devnode "^(ram|raw|loop|fd|md|dm-|sr|scd|st)[0-9]*"
        devnode "^hd[a-z]"
        devnode "^cciss.*"
}
```

Step 6: Customize multipath parameters for ONTAP LUNs

If your host is connected to LUNs from other vendors and any of the multipath parameter settings are overridden, you need to correct them by adding stanzas later in the `multipath.conf` file that apply specifically to ONTAP LUNs. If you don't do this, the ONTAP LUNs might not work as expected.

Check your `/etc/multipath.conf` file, especially in the defaults section, for settings that might be overriding the default settings for multipath parameters.

> You shouldn't override the recommended parameter settings for ONTAP LUNs. These settings are required for optimal performance of your host configuration. Contact NetApp support, your OS vendor, or both for more information.

The following example shows how to correct an overridden default. In this example, the `multipath.conf` file defines values for `path_checker` and `no_path_retry` that aren't compatible with ONTAP LUNs, and you can't remove these parameters because ONTAP storage arrays are still attached to the host. Instead, you correct the values for `path_checker` and `no_path_retry` by adding a device stanza to the `multipath.conf` file that applies specifically to the ONTAP LUNs.

Show example

```
defaults {
   path_checker       readsector0
   no_path_retry      fail
}

devices {
   device {
      vendor           "NETAPP"
      product          "LUN"
      no_path_retry    queue
      path_checker     tur
   }
}
```

Step 7: Review the known issues

There are no known issues.

What's next?

- Learn about using the Linux Host Utilities tool.
- Learn about ASM mirroring

  Automatic Storage Management (ASM) mirroring might require changes to the Linux multipath settings to allow ASM to recognize a problem and switch over to an alternate failure group. Most ASM configurations on ONTAP use external redundancy, which means that data protection is provided by the external array and ASM doesn't mirror data. Some sites use ASM with normal redundancy to provide two-way mirroring, normally across different sites. See Oracle databases on ONTAP for further information.

- Learn about SUSE Linux Virtualization (KVM)

  SUSE Linux can serve as a KVM host. This enables you to run multiple virtual machines on a single physical server using the Linux Kernel-based Virtual Machine (KVM) technology. The KVM host doesn't require explicit host configuration settings for ONTAP LUNs.

Configure SUSE Linux Enterprise Server 15 SPx for FCP and iSCSI with ONTAP storage

The Linux Host Utilities software provides management and diagnostic tools for Linux hosts that are connected to ONTAP storage. When you install the Linux Host Utilities on a SUSE Linux Enterprise Server 15 SPx host, you can use the Host Utilities to help you manage FCP and iSCSI protocol operations with ONTAP LUNs.

Step 1: Optionally, enable SAN booting

You can configure your host to use SAN booting to simplify deployment and improve scalability.

Before you begin

Use the Interoperability Matrix Tool to verify that your Linux OS, host bus adapter (HBA), HBA firmware, HBA boot BIOS, and ONTAP version support SAN booting.

Steps

1. Create a SAN boot LUN and map it to the host.
2. Enable SAN booting in the server BIOS for the ports to which the SAN boot LUN is mapped.

   For information on how to enable the HBA BIOS, see your vendor-specific documentation.

3. Verify that the configuration was successful by rebooting the host and verifying that the OS is up and running.

Step 2: Install the Linux Host Utilities

NetApp strongly recommends installing the Linux Host Utilities to support ONTAP LUN management and assist technical support with gathering configuration data.

Install Linux Host Utilities 8.0.

Installing the Linux Host Utilities doesn't change any host timeout settings on your Linux host.

Step 3: Confirm the multipath configuration for your host

You can use multipathing with SUSE Linux Enterprise Server 15 SPx to manage ONTAP LUNs.

To ensure that multipathing is configured correctly for your host, verify that the `/etc/multipath.conf` file is defined and that you have the NetApp recommended settings configured for your ONTAP LUNs.

Steps

1. Verify that the `/etc/multipath.conf` file exists:

   ```
   ls /etc/multipath.conf
   ```

   If the file doesn't exist, create an empty, zero-byte file:

   ```
   touch /etc/multipath.conf
   ```

2. The first time the `multipath.conf` file is created, you might need to enable and start the multipath services to load the recommended settings:

   ```
   systemctl enable multipathd
   ```

   ```
   systemctl start multipathd
   ```

3. Each time you boot the host, the empty `/etc/multipath.conf` zero-byte file automatically loads the NetApp recommended host multipath parameters as the default settings. You shouldn't need to make changes to the `/etc/multipath.conf` file for your host because the host operating system is compiled with the multipath parameters that recognize and manage ONTAP LUNs correctly.

   The following table shows the native Linux OS compiled multipath parameter settings for ONTAP LUNs.

Show parameter settings

Parameter	Setting
detect_prio	yes
dev_loss_tmo	"infinity"
failback	immediate
fast_io_fail_tmo	5
features	"2 pg_init_retries 50"
flush_on_last_del	"yes"
hardware_handler	"0"
no_path_retry	queue
path_checker	"tur"
path_grouping_policy	"group_by_prio"
path_selector	"service-time 0"
polling_interval	5
prio	"ontap"
product	LUN
retain_attached_hw_handler	yes
rr_weight	"uniform"
user_friendly_names	no
vendor	NETAPP

4. Verify the parameter settings and path status for your ONTAP LUNs:

```
multipath -ll
```

The default multipath parameters support ASA, AFF, and FAS configurations. In these configurations, a single ONTAP LUN shouldn’t require more than four paths. Having more than four paths can cause problems during a storage failure.

The following example outputs show the correct parameter settings and path status for ONTAP LUNs in an ASA, AFF, or FAS configuration.

ASA configuration

An ASA configuration optimizes all paths to a given LUN, keeping them active. This improves performance by serving I/O operations through all paths at the same time.

Show example

```
# multipath -ll
3600a09803831347657244e527766394e dm-5 NETAPP,LUN C-Mode
size=80G features='3 queue_if_no_path pg_init_retries 50'
hwhandler='1 alua' wp=rw
`-+- policy='service-time 0' prio=50 status=active
  |- 3:0:7:9    sdco 69:192  active ready running
  |- 3:0:8:9    sddi 71:0    active ready running
  |- 14:0:8:9   sdjq 65:320  active ready running
  `- 14:0:7:9   sdiw 8:256   active ready running
```

AFF or FAS configuration

An AFF or FAS configuration should have two groups of paths with higher and lower priorities. Higher priority Active/Optimized paths are served by the controller where the aggregate is located. Lower priority paths are active but non-optimized because they are served by a different controller. Non-optimized paths are only used when optimized paths aren’t available.

The following example displays the output for an ONTAP LUN with two Active/Optimized paths and two Active/Non-Optimized paths:

Show example

```
# multipath -ll
3600a09803831347657244e527766394e dm-5 NETAPP,LUN C-Mode
size=80G features='3 queue_if_no_path pg_init_retries 50'
hwhandler='1 alua' wp=rw
|-+- policy='service-time 0' prio=50 status=active
| |- 3:0:3:0    sdd  8:48    active ready running
| |- 3:0:4:0    sdx  65:112  active ready running
`-+- policy='service-time 0' prio=10 status=enabled
  |- 14:0:2:0   sdfk 130:96  active ready running
  `- 14:0:5:0   sdgz 132:240 active ready running
```

Step 4: Confirm the iSCSI configuration for your host

Ensure that iSCSI is configured correctly for your host.

About this task

You perform the following steps on the iSCSI host.

Steps

1. Verify that the iSCSI initiator package (iscsi-initiator-utils) is installed:

   ```
   rpm -qa | grep open-iscsi
   ```

   You should see an output similar to the following example:

   ```
   open-iscsi-2.1.11-160000.2.2.x86_64
   ```

2. Verify the iSCSI initiator node name, which is located in the `/etc/iscsi/initiatorname.iscsi` file:

   ```
   InitiatorName=iqn.YYYY-MM.com.<vendor>:<host_name>
   ```

3. Configure the iSCSI session timeout parameter located in the `/etc/iscsi/iscsid.conf` file:

   ```
   node.session.timeo.replacement_timeout = 5
   ```

   The iSCSI `replacement_timeout` parameter controls how long the iSCSI layer should wait for a timed-out path or session to reestablish itself before failing any commands on it. You should set the value of `replacement_timeout` to 5 in the iSCSI configuration file.

4. Enable the iSCSI service:

   ```
   $systemctl enable iscsid
   ```

5. Start the iSCSI service:

   ```
   $systemctl start iscsid
   ```

6. Verify that the iSCSI service is running:

   ```
   $systemctl status iscsid
   ```

Show example

```
● iscsid.service - Open-iSCSI
     Loaded: loaded (/usr/lib/systemd/system/iscsid.service;
enabled; preset: disabled)
     Active: active (running) since Tue 2025-12-02 11:36:21 EST; 2
weeks 1 day ago
TriggeredBy: ● iscsid.socket
       Docs: man:iscsid(8)
             man:iscsiuio(8)
             man:iscsiadm(8)
   Main PID: 2263 (iscsid)
     Status: "Ready to process requests"
      Tasks: 1 (limit: 816061)
     Memory: 18.5M
        CPU: 14.480s
     CGroup: /system.slice/iscsid.service
             └─2263 /usr/sbin/iscsid -f -d2
```

7. Discover the iSCSI targets:

```
$iscsiadm --mode discovery --op update --type sendtargets --portal
<target_IP>
```

show example

```
iscsiadm --mode discovery --op update --type sendtargets --portal
192.168.30.87
192.168.30.87:3260,1139 iqn.1992-
08.com.netapp:sn.064a9b19b3ee11f09dcad039eabac370:vs.23
192.168.31.97:3260,1142 iqn.1992-
08.com.netapp:sn.064a9b19b3ee11f09dcad039eabac370:vs.23
192.168.31.87:3260,1141 iqn.1992-
08.com.netapp:sn.064a9b19b3ee11f09dcad039eabac370:vs.23
192.168.30.97:3260,1140 iqn.1992-
08.com.netapp:sn.064a9b19b3ee11f09dcad039eabac370:vs.23
```

8. Log in to the targets:

```
$iscsiadm --mode node -l all
```

9. Set iSCSI to log in automatically when the host boots:

```
$iscsiadm --mode node -T <target_name> -p <ip:port> -o update -n
node.startup -v automatic
```

You should see an output similar to the following example:

```
iscsiadm --mode node -T iqn.1992-
08.com.netapp:sn.064a9b19b3ee11f09dcad039eabac370:vs.23 -p
192.168.30.87:3260 -o update -n node.startup -v automatic
```

10. Verify the iSCSI sessions:

```
$iscsiadm --mode session
```

Show example

```
iscsiadm --mode session
tcp: [1] 192.168.30.87:3260,1139 iqn.1992-
08.com.netapp:sn.064a9b19b3ee11f09dcad039eabac370:vs.23 (non-flash)
tcp: [2] 192.168.31.97:3260,1142 iqn.1992-
08.com.netapp:sn.064a9b19b3ee11f09dcad039eabac370:vs.23 (non-flash)
tcp: [3] 192.168.31.87:3260,1141 iqn.1992-
08.com.netapp:sn.064a9b19b3ee11f09dcad039eabac370:vs.23 (non-flash)
tcp: [4] 192.168.30.97:3260,1140 iqn.1992-
08.com.netapp:sn.064a9b19b3ee11f09dcad039eabac370:vs.23 (non-flash)
```

Step 5: Optionally, exclude a device from multipathing

If required, you can exclude a device from multipathing by adding the WWID for the unwanted device to the "blacklist" stanza for the `multipath.conf` file.

Steps

1. Determine the WWID:

```
/lib/udev/scsi_id -gud /dev/sda
```

"sda" is the local SCSI disk that you want to add to the blacklist.

An example WWID is `360030057024d0730239134810c0cb833`.

2. Add the WWID to the "blacklist" stanza:

```
blacklist {
         wwid   360030057024d0730239134810c0cb833
        devnode "^(ram|raw|loop|fd|md|dm-|sr|scd|st)[0-9]*"
        devnode "^hd[a-z]"
        devnode "^cciss.*"
}
```

Step 6: Customize multipath parameters for ONTAP LUNs

If your host is connected to LUNs from other vendors and any of the multipath parameter settings are overridden, you need to correct them by adding stanzas later in the `multipath.conf` file that apply specifically to ONTAP LUNs. If you don't do this, the ONTAP LUNs might not work as expected.

Check your `/etc/multipath.conf` file, especially in the defaults section, for settings that might be overriding the default settings for multipath parameters.

> ⚠ You shouldn't override the recommended parameter settings for ONTAP LUNs. These settings are required for optimal performance of your host configuration. Contact NetApp support, your OS vendor, or both for more information.

The following example shows how to correct an overridden default. In this example, the `multipath.conf` file defines values for `path_checker` and `no_path_retry` that aren't compatible with ONTAP LUNs, and you can't remove these parameters because ONTAP storage arrays are still attached to the host. Instead, you correct the values for `path_checker` and `no_path_retry` by adding a device stanza to the `multipath.conf` file that applies specifically to the ONTAP LUNs.

Show example

```
defaults {
   path_checker       readsector0
   no_path_retry      fail
}

devices {
   device {
      vendor           "NETAPP"
      product          "LUN"
      no_path_retry    queue
      path_checker     tur
   }
}
```

Step 7: Review the known issues

SUSE Linux Enterprise Server 15 SPx with ONTAP storage has the following known issues.

15 SP1

NetApp Bug ID	Title	Description
1246622	Remote ports transit to a blocked state on SLES15SP1 with Emulex LPe12002 8GB FC during storage failover operations.	Remote ports transit to a blocked state on SLES15SP1 with Emulex LPe12002 8GB Fibre Channel (FC) during storage failover operations. When the storage node returns to an optimal state, the LIFs also come up and the remote port state should read "online." Occasionally, the remote port state might continue to read as "blocked" or "not present." This state can lead to a "failed faulty" path to LUNs at the multipath layer as well as an I/O outage for those LUNs. You can check the remoteport's details against the following sample commands: ---- cat/sys/class/fc_host/host*/device/rport*/fc_remote_ports/rport*/port_name cat/sys/class/fc_host/host*/device/rport*/fc_remote_ports/rport*/port_state ----

15

NetApp Bug ID	Title	Description
1154309	SLES 15 host with more than 20 mapped LUNs might go into maintenance mode after a reboot	SLES 15 host with more than 20 mapped LUNs might go into maintenance mode after a reboot. The maintenance mode becomes single user mode following the message: `Give root password for maintenance (or press Control-D to continue)`

What's next?

- Learn about using the Linux Host Utilities tool.
- Learn about ASM mirroring

  Automatic Storage Management (ASM) mirroring might require changes to the Linux multipath settings to allow ASM to recognize a problem and switch over to an alternate failure group. Most ASM configurations on ONTAP use external redundancy, which means that data protection is provided by the external array and ASM doesn't mirror data. Some sites use ASM with normal redundancy to provide two-way mirroring, normally across different sites. See Oracle databases on ONTAP for further information.

- Learn about SUSE Linux Virtualization (KVM)

  SUSE Linux can serve as a KVM host. This enables you to run multiple virtual machines on a single

physical server using the Linux Kernel-based Virtual Machine (KVM) technology. The KVM host doesn't require explicit host configuration settings for ONTAP LUNs.

Ubuntu

Configure Ubuntu 24.04 for FCP and iSCSI with ONTAP storage

The Linux Host Utilities software provides management and diagnostic tools for Linux hosts that are connected to ONTAP storage. When you install the Linux Host Utilities on an Ubuntu 24.04 host, you can use the Host Utilities to help you manage FCP and iSCSI protocol operations with ONTAP LUNs.

Step 1: Optionally, enable SAN booting

You can configure your host to use SAN booting to simplify deployment and improve scalability.

Before you begin

Use the Interoperability Matrix Tool to verify that your Linux OS, host bus adapter (HBA), HBA firmware, HBA boot BIOS, and ONTAP version support SAN booting.

Steps

1. Create a SAN boot LUN and map it to the host.
2. Enable SAN booting in the server BIOS for the ports to which the SAN boot LUN is mapped.

   For information on how to enable the HBA BIOS, see your vendor-specific documentation.

3. Verify that the configuration was successful by rebooting the host and verifying that the OS is up and running.

Step 2: Install the Linux Host Utilities

NetApp strongly recommends installing the Linux Host Utilities to support ONTAP LUN management and assist technical support with gathering configuration data.

Install Linux Host Utilities 8.0.

Installing the Linux Host Utilities doesn't change any host timeout settings on your Linux host.

Step 3: Confirm the multipath configuration for your host

You can use multipathing with Ubuntu 24.04 to manage ONTAP LUNs.

To ensure that multipathing is configured correctly for your host, verify that the `/etc/multipath.conf` file is defined and that you have the NetApp recommended settings configured for your ONTAP LUNs.

Steps

1. Verify that the `/etc/multipath.conf` file exits. If the file doesn't exist, create an empty, zero-byte file:

```
touch /etc/multipath.conf
```

2. The first time the `multipath.conf` file is created, you might need to enable and start the multipath services to load the recommended settings:

```
systemctl enable multipathd
```

```
systemctl start multipathd
```

3. Each time you boot the host, the empty `/etc/multipath.conf` zero-byte file automatically loads the NetApp recommended host multipath parameters as the default settings. You shouldn't need to make changes to the `/etc/multipath.conf` file for your host because the operating system is compiled with the multipath parameters that recognize and manage ONTAP LUNs correctly.

   The following table shows the Linux OS native compiled multipath parameter settings for ONTAP LUNs.

   Show parameter settings

Parameter	Setting
detect_prio	yes
dev_loss_tmo	"infinity"
failback	immediate
fast_io_fail_tmo	5
features	"2 pg_init_retries 50"
flush_on_last_del	"yes"
hardware_handler	"0"
no_path_retry	queue
path_checker	"tur"
path_grouping_policy	"group_by_prio"
path_selector	"service-time 0"
polling_interval	5
prio	"ontap"
product	LUN
retain_attached_hw_handler	yes
rr_weight	"uniform"
user_friendly_names	no
vendor	NETAPP

4. Verify the parameter settings and path status for your ONTAP LUNs:

```
multipath -ll
```

The default multipath parameters support ASA, AFF, and FAS configurations. In these configurations, a single ONTAP LUN shouldn’t require more than four paths. Having more than four paths can cause problems during a storage failure.

The following example outputs show the correct parameter settings and path status for ONTAP LUNs in an ASA, AFF, or FAS configuration.

ASA configuration

An ASA configuration optimizes all paths to a given LUN, keeping them active. This improves performance by serving I/O operations through all paths at the same time.

Show example

```
# multipath -ll
3600a0980383145595533f524d6c652f62 dm-24 NETAPP,LUN C-Mode
size=10G features='3 queue_if_no_path pg_init_retries 50'
hwhandler='1 alua' wp=rw
`-+- policy='service-time 0' prio=50 status=active
  |- 11:0:1:13 sdm  8:192  active ready running
  |- 11:0:3:13 sdah 66:16  active ready running
  |- 12:0:1:13 sdbc 67:96  active ready running
  `- 12:0:3:13 sdbx 68:176 active ready running
```

AFF or FAS configuration

An AFF or FAS configuration should have two groups of paths with higher and lower priorities. Higher priority Active/Optimized paths are served by the controller where the aggregate is located. Lower priority paths are active but non-optimized because they are served by a different controller. Non-optimized paths are only used when optimized paths aren't available.

The following example displays the output for an ONTAP LUN with two Active/Optimized paths and two Active/Non-Optimized paths:

Show example

```
# multipath -ll
3600a0980383148373524536946b542f4a dm-0 NETAPP,LUN C-Mode
size=160G features='3 queue_if_no_path pg_init_retries 50'
hwhandler='1 alua' wp=rw
|-+- policy='service-time 0' prio=50 status=active
| |- 14:0:3:0 sdbk 67:224 active ready running
| `- 15:0:2:0 sdbl 67:240 active ready running
`-+- policy='service-time 0' prio=10 status=enabled
  |- 14:0:0:0 sda  8:0    active ready running
  `- 15:0:1:0 sdv  65:80  active ready running
```

Step 4: Confirm the iSCSI configuration for your host

Ensure that iSCSI is configured correctly for your host.

About this task

You perform the following steps on the iSCSI host.

Steps

1. Verify that the iSCSI initiator package (open-iscsi) is installed:

   ```
   $apt list |grep open-iscsi
   ```

   You should see an output similar to the following example:

   ```
   open-iscsi/noble-updates,noble-updates,now 2.1.9-3ubuntu5.4 amd64
   ```

2. Verify the iSCSI initiator node name, which is located in the `/etc/iscsi/initiatorname.iscsi` file:

   ```
   InitiatorName=iqn.YYYY-MM.com.<vendor>:<host_name>
   ```

3. Configure the iSCSI session timeout parameter located in the `/etc/iscsi/iscsid.conf` file:

   ```
   node.session.timeo.replacement_timeout = 5
   ```

   The iSCSI `replacement_timeout` parameter controls how long the iSCSI layer should wait for a timed-out path or session to reestablish itself before failing any commands on it. You should set the value of `replacement_timeout` to 5 in the iSCSI configuration file.

4. Enable the iSCSI service:

   ```
   $systemctl enable iscsid
   ```

5. Start the iSCSI service:

   ```
   $systemctl start iscsid
   ```

6. Verify that the iSCSI service is running:

   ```
   $systemctl status iscsid
   ```

Show example

```
●iscsid.service - iSCSI initiator daemon (iscsid)
     Loaded: loaded (/usr/lib/systemd/system/iscsid.service;
enabled; preset: disabled)
     Active: active (running) since Mon 2026-01-12 12:53:18 IST; 2
days ago
TriggeredBy: ● iscsid.socket
       Docs: man:iscsid(8)
   Main PID: 1127419 (iscsid)
      Tasks: 2 (limit: 76557)
     Memory: 4.3M (peak: 8.8M)
        CPU: 1.657s
     CGroup: /system.slice/iscsid.service
             ├─1127418 /usr/sbin/iscsid
             └─1127419 /usr/sbin/iscsid
```

7. Discover the iSCSI targets:

```
$iscsiadm --mode discovery --op update --type sendtargets --portal
<target_IP>
```

show example

```
iscsiadm --mode discovery --op update --type sendtargets --portal
192.168.100.197
192.168.100.197:3260,1046 iqn.1992-
08.com.netapp:sn.7cd154a7d35411f0a25ed039eaa95f59:vs.8
192.168.200.199:3260,1049 iqn.1992-
08.com.netapp:sn.7cd154a7d35411f0a25ed039eaa95f59:vs.8
192.168.100.199:3260,1048 iqn.1992-
08.com.netapp:sn.7cd154a7d35411f0a25ed039eaa95f59:vs.8
192.168.200.197:3260,1047 iqn.1992-
08.com.netapp:sn.7cd154a7d35411f0a25ed039eaa95f59:vs.8
```

8. Log in to the targets:

```
$iscsiadm --mode node -l all
```

9. Set iSCSI to log in automatically when the host boots:

```
$iscsiadm --mode node -T <target_name> -p <ip:port> -o update -n
node.startup -v automatic
```

You should see an output similar to the following example:

```
iscsiadm --mode node -T iqn.1992-
08.com.netapp:sn.7cd154a7d35411f0a25ed039eaa95f59:vs.8 -p
192.168.100.197:3260 -o update -n node.startup -v automatic
```

10. Verify the iSCSI sessions:

```
$iscsiadm --mode session
```

Show example

```
iscsiadm --mode session
tcp: [1] 192.168.200.197:3260,1047 iqn.1992-
08.com.netapp:sn.7cd154a7d35411f0a25ed039eaa95f59:vs.8 (non-flash)
tcp: [2] 192.168.100.197:3260,1046 iqn.1992-
08.com.netapp:sn.7cd154a7d35411f0a25ed039eaa95f59:vs.8 (non-flash)
tcp: [3] 192.168.100.199:3260,1048 iqn.1992-
08.com.netapp:sn.7cd154a7d35411f0a25ed039eaa95f59:vs.8 (non-flash)
tcp: [4] 192.168.200.199:3260,1049 iqn.1992-
08.com.netapp:sn.7cd154a7d35411f0a25ed039eaa95f59:vs.8 (non-flash)
```

Step 5: Optionally, exclude a device from multipathing

If required, you can exclude a device from multipathing by adding the WWID for the unwanted device to the "blacklist" stanza for the `multipath.conf` file.

Steps

1. Determine the WWID:

```
/lib/udev/scsi_id -gud /dev/sda
```

"sda" is the local SCSI disk that you want to add to the blacklist.

An example WWID is `360030057024d0730239134810c0cb833`.

2. Add the WWID to the "blacklist" stanza:

```
blacklist {
         wwid   360030057024d0730239134810c0cb833
        devnode "^(ram|raw|loop|fd|md|dm-|sr|scd|st)[0-9]*"
        devnode "^hd[a-z]"
        devnode "^cciss.*"
}
```

Step 6: Customize multipath parameters for ONTAP LUNs

If your host is connected to LUNs from other vendors and any of the multipath parameter settings are overridden, you need to correct them by adding stanzas later in the `multipath.conf` file that apply specifically to ONTAP LUNs. If you don't do this, the ONTAP LUNs might not work as expected.

Check your `/etc/multipath.conf` file, especially in the defaults section, for settings that might be overriding the default settings for multipath parameters.

> ⚠ You shouldn't override the recommended parameter settings for ONTAP LUNs. These settings are required for optimal performance of your host configuration. Contact NetApp support, your OS vendor, or both for more information.

The following example shows how to correct an overridden default. In this example, the `multipath.conf` file defines values for `path_checker` and `no_path_retry` that aren't compatible with ONTAP LUNs, and you can't remove these parameters because ONTAP storage arrays are still attached to the host. Instead, you correct the values for `path_checker` and `no_path_retry` by adding a device stanza to the `multipath.conf` file that applies specifically to the ONTAP LUNs.

Show example

```
defaults {
   path_checker       readsector0
   no_path_retry      fail
}

devices {
   device {
      vendor           "NETAPP"
      product          "LUN"
      no_path_retry    queue
      path_checker     tur
   }
}
```

Step 7: Review the known issues

There are no known issues.

What's next?

- Learn about using the Linux Host Utilities tool.
- Learn about ASM mirroring

  Automatic Storage Management (ASM) mirroring might require changes to the Linux multipath settings to allow ASM to recognize a problem and switch over to an alternate failure group. Most ASM configurations on ONTAP use external redundancy, which means that data protection is provided by the external array and ASM doesn't mirror data. Some sites use ASM with normal redundancy to provide two-way mirroring, normally across different sites. See Oracle databases on ONTAP for further information.

- Learn about Ubuntu Linux Virtualization (KVM)

  Ubuntu Linux can serve as a KVM host. This enables you to run multiple virtual machines on a single physical server using the Linux Kernel-based Virtual Machine (KVM) technology. The KVM host doesn't require explicit host configuration settings for ONTAP LUNs.

Configure Ubuntu 22.04 for FCP and iSCSI with ONTAP storage

The Linux Host Utilities software provides management and diagnostic tools for Linux hosts that are connected to ONTAP storage. When you install the Linux Host Utilities on an Ubuntu 22.04 host, you can use the Host Utilities to help you manage FCP and iSCSI protocol operations with ONTAP LUNs.

Step 1: Optionally, enable SAN booting

You can configure your host to use SAN booting to simplify deployment and improve scalability.

Before you begin

Use the Interoperability Matrix Tool to verify that your Linux OS, host bus adapter (HBA), HBA firmware, HBA boot BIOS, and ONTAP version support SAN booting.

Steps

1. Create a SAN boot LUN and map it to the host.
2. Enable SAN booting in the server BIOS for the ports to which the SAN boot LUN is mapped.

   For information on how to enable the HBA BIOS, see your vendor-specific documentation.

3. Verify that the configuration was successful by rebooting the host and verifying that the OS is up and running.

Step 2: Install the Linux Host Utilities

NetApp strongly recommends installing the Linux Host Utilities to support ONTAP LUN management and assist technical support with gathering configuration data.

Install Linux Host Utilities 8.0.

Installing the Linux Host Utilities doesn't change any host timeout settings on your Linux host.

Step 3: Confirm the multipath configuration for your host

You can use multipathing with Ubuntu 22.04 to manage ONTAP LUNs.

To ensure that multipathing is configured correctly for your host, verify that the `/etc/multipath.conf` file is defined and that you have the NetApp recommended settings configured for your ONTAP LUNs.

Steps

1. Verify that the `/etc/multipath.conf` file exits. If the file doesn't exist, create an empty, zero-byte file:

```
touch /etc/multipath.conf
```

2. The first time the `multipath.conf` file is created, you might need to enable and start the multipath services to load the recommended settings:

```
systemctl enable multipathd
```

```
systemctl start multipathd
```

3. Each time you boot the host, the empty `/etc/multipath.conf` zero-byte file automatically loads the NetApp recommended host multipath parameters as the default settings. You shouldn't need to make changes to the `/etc/multipath.conf` file for your host because the operating system is compiled with the multipath parameters that recognize and manage ONTAP LUNs correctly.

   The following table shows the Linux OS native compiled multipath parameter settings for ONTAP LUNs.

Show parameter settings

Parameter	Setting
detect_prio	yes
dev_loss_tmo	"infinity"
failback	immediate
fast_io_fail_tmo	5
features	"2 pg_init_retries 50"
flush_on_last_del	"yes"
hardware_handler	"0"
no_path_retry	queue
path_checker	"tur"
path_grouping_policy	"group_by_prio"
path_selector	"service-time 0"
polling_interval	5
prio	"ontap"
product	LUN
retain_attached_hw_handler	yes
rr_weight	"uniform"
user_friendly_names	no
vendor	NETAPP

4. Verify the parameter settings and path status for your ONTAP LUNs:

```
multipath -ll
```

The default multipath parameters support ASA, AFF, and FAS configurations. In these configurations, a single ONTAP LUN shouldn’t require more than four paths. Having more than four paths can cause problems during a storage failure.

The following example outputs show the correct parameter settings and path status for ONTAP LUNs in an ASA, AFF, or FAS configuration.

ASA configuration

An ASA configuration optimizes all paths to a given LUN, keeping them active. This improves performance by serving I/O operations through all paths at the same time.

Show example

```
# multipath -ll
3600a0980383145595533f524d6c652f62 dm-24 NETAPP,LUN C-Mode
size=10G features='3 queue_if_no_path pg_init_retries 50'
hwhandler='1 alua' wp=rw
`-+- policy='service-time 0' prio=50 status=active
  |- 11:0:1:13 sdm  8:192  active ready running
  |- 11:0:3:13 sdah 66:16  active ready running
  |- 12:0:1:13 sdbc 67:96  active ready running
  `- 12:0:3:13 sdbx 68:176 active ready running
```

AFF or FAS configuration

An AFF or FAS configuration should have two groups of paths with higher and lower priorities. Higher priority Active/Optimized paths are served by the controller where the aggregate is located. Lower priority paths are active but non-optimized because they are served by a different controller. Non-optimized paths are only used when optimized paths aren’t available.

The following example displays the output for an ONTAP LUN with two Active/Optimized paths and two Active/Non-Optimized paths:

Show example

```
# multipath -ll
3600a0980383148373524536946b542f4a dm-0 NETAPP,LUN C-Mode
size=160G features='3 queue_if_no_path pg_init_retries 50'
hwhandler='1 alua' wp=rw
|-+- policy='service-time 0' prio=50 status=active
| |- 14:0:3:0 sdbk 67:224 active ready running
| `- 15:0:2:0 sdbl 67:240 active ready running
`-+- policy='service-time 0' prio=10 status=enabled
  |- 14:0:0:0 sda  8:0    active ready running
  `- 15:0:1:0 sdv  65:80  active ready running
```

Step 4: Confirm the iSCSI configuration for your host

Ensure that iSCSI is configured correctly for your host.

About this task

You perform the following steps on the iSCSI host.

Steps

1. Verify that the iSCSI initiator package (open-iscsi) is installed:

   ```
   $apt list |grep open-iscsi
   ```

   You should see an output similar to the following example:

   ```
   open-iscsi/noble-updates,noble-updates,now 2.1.9-3ubuntu5.4 amd64
   ```

2. Verify the iSCSI initiator node name, which is located in the `/etc/iscsi/initiatorname.iscsi` file:

   ```
   InitiatorName=iqn.YYYY-MM.com.<vendor>:<host_name>
   ```

3. Configure the iSCSI session timeout parameter located in the `/etc/iscsi/iscsid.conf` file:

   ```
   node.session.timeo.replacement_timeout = 5
   ```

   The iSCSI `replacement_timeout` parameter controls how long the iSCSI layer should wait for a timed-out path or session to reestablish itself before failing any commands on it. You should set the value of `replacement_timeout` to 5 in the iSCSI configuration file.

4. Enable the iSCSI service:

   ```
   $systemctl enable iscsid
   ```

5. Start the iSCSI service:

   ```
   $systemctl start iscsid
   ```

6. Verify that the iSCSI service is running:

   ```
   $systemctl status iscsid
   ```

Show example

```
●iscsid.service - iSCSI initiator daemon (iscsid)
     Loaded: loaded (/usr/lib/systemd/system/iscsid.service;
enabled; preset: disabled)
     Active: active (running) since Mon 2026-01-12 12:53:18 IST; 2
days ago
TriggeredBy: ● iscsid.socket
       Docs: man:iscsid(8)
   Main PID: 1127419 (iscsid)
      Tasks: 2 (limit: 76557)
     Memory: 4.3M (peak: 8.8M)
        CPU: 1.657s
     CGroup: /system.slice/iscsid.service
             ├─1127418 /usr/sbin/iscsid
             └─1127419 /usr/sbin/iscsid
```

7. Discover the iSCSI targets:

```
$iscsiadm --mode discovery --op update --type sendtargets --portal
<target_IP>
```

show example

```
iscsiadm --mode discovery --op update --type sendtargets --portal
192.168.100.197
192.168.100.197:3260,1046 iqn.1992-
08.com.netapp:sn.7cd154a7d35411f0a25ed039eaa95f59:vs.8
192.168.200.199:3260,1049 iqn.1992-
08.com.netapp:sn.7cd154a7d35411f0a25ed039eaa95f59:vs.8
192.168.100.199:3260,1048 iqn.1992-
08.com.netapp:sn.7cd154a7d35411f0a25ed039eaa95f59:vs.8
192.168.200.197:3260,1047 iqn.1992-
08.com.netapp:sn.7cd154a7d35411f0a25ed039eaa95f59:vs.8
```

8. Log in to the targets:

```
$iscsiadm --mode node -l all
```

9. Set iSCSI to log in automatically when the host boots:

```
$iscsiadm --mode node -T <target_name> -p <ip:port> -o update -n
node.startup -v automatic
```

You should see an output similar to the following example:

```
iscsiadm --mode node -T iqn.1992-
08.com.netapp:sn.7cd154a7d35411f0a25ed039eaa95f59:vs.8 -p
192.168.100.197:3260 -o update -n node.startup -v automatic
```

10. Verify the iSCSI sessions:

```
$iscsiadm --mode session
```

Show example

```
iscsiadm --mode session
tcp: [1] 192.168.200.197:3260,1047 iqn.1992-
08.com.netapp:sn.7cd154a7d35411f0a25ed039eaa95f59:vs.8 (non-flash)
tcp: [2] 192.168.100.197:3260,1046 iqn.1992-
08.com.netapp:sn.7cd154a7d35411f0a25ed039eaa95f59:vs.8 (non-flash)
tcp: [3] 192.168.100.199:3260,1048 iqn.1992-
08.com.netapp:sn.7cd154a7d35411f0a25ed039eaa95f59:vs.8 (non-flash)
tcp: [4] 192.168.200.199:3260,1049 iqn.1992-
08.com.netapp:sn.7cd154a7d35411f0a25ed039eaa95f59:vs.8 (non-flash)
```

Step 5: Optionally, exclude a device from multipathing

If required, you can exclude a device from multipathing by adding the WWID for the unwanted device to the "blacklist" stanza for the `multipath.conf` file.

Steps

1. Determine the WWID:

```
/lib/udev/scsi_id -gud /dev/sda
```

"sda" is the local SCSI disk that you want to add to the blacklist.

An example WWID is `360030057024d0730239134810c0cb833`.

2. Add the WWID to the "blacklist" stanza:

```
blacklist {
         wwid   360030057024d0730239134810c0cb833
        devnode "^(ram|raw|loop|fd|md|dm-|sr|scd|st)[0-9]*"
        devnode "^hd[a-z]"
        devnode "^cciss.*"
}
```

Step 6: Customize multipath parameters for ONTAP LUNs

If your host is connected to LUNs from other vendors and any of the multipath parameter settings are overridden, you need to correct them by adding stanzas later in the `multipath.conf` file that apply specifically to ONTAP LUNs. If you don't do this, the ONTAP LUNs might not work as expected.

Check your `/etc/multipath.conf` file, especially in the defaults section, for settings that might be overriding the default settings for multipath parameters.

> ⚠ You shouldn't override the recommended parameter settings for ONTAP LUNs. These settings are required for optimal performance of your host configuration. Contact NetApp support, your OS vendor, or both for more information.

The following example shows how to correct an overridden default. In this example, the `multipath.conf` file defines values for `path_checker` and `no_path_retry` that aren't compatible with ONTAP LUNs, and you can't remove these parameters because ONTAP storage arrays are still attached to the host. Instead, you correct the values for `path_checker` and `no_path_retry` by adding a device stanza to the `multipath.conf` file that applies specifically to the ONTAP LUNs.

Show example

```
defaults {
   path_checker       readsector0
   no_path_retry      fail
}

devices {
   device {
      vendor           "NETAPP"
      product          "LUN"
      no_path_retry    queue
      path_checker     tur
   }
}
```

Step 7: Review the known issues

There are no known issues.

What's next?

- Learn about using the Linux Host Utilities tool.
- Learn about ASM mirroring

  Automatic Storage Management (ASM) mirroring might require changes to the Linux multipath settings to allow ASM to recognize a problem and switch over to an alternate failure group. Most ASM configurations on ONTAP use external redundancy, which means that data protection is provided by the external array and ASM doesn't mirror data. Some sites use ASM with normal redundancy to provide two-way mirroring, normally across different sites. See Oracle databases on ONTAP for further information.

- Learn about Ubuntu Linux Virtualization (KVM)

  Ubuntu Linux can serve as a KVM host. This enables you to run multiple virtual machines on a single physical server using the Linux Kernel-based Virtual Machine (KVM) technology. The KVM host doesn't require explicit host configuration settings for ONTAP LUNs.

Configure Ubuntu 20.04 for FCP and iSCSI with ONTAP storage

Configure Ubuntu 20.04 for multipathing and with specific parameters and settings for FCP and iSCSI protocol operations with ONTAP storage.

> The Linux Host Utilities software package doesn't support Ubuntu operating systems.
>
> You don't need to manually configure Kernel-based Virtual Machine (KVM) settings because ONTAP LUNs are automatically mapped to the hypervisor.

Step 1: Optionally, enable SAN booting

You can configure your host to use SAN booting to simplify deployment and improve scalability.

Before you begin

Use the Interoperability Matrix Tool to verify that your Linux OS, host bus adapter (HBA), HBA firmware, HBA boot BIOS, and ONTAP version support SAN booting.

Steps

1. Create a SAN boot LUN and map it to the host.
2. Enable SAN booting in the server BIOS for the ports to which the SAN boot LUN is mapped.

   For information on how to enable the HBA BIOS, see your vendor-specific documentation.

3. Verify that the configuration was successful by rebooting the host and verifying that the OS is up and running.

Step 2: Confirm the multipath configuration for your host

You can use multipathing with Ubuntu 20.04 to manage ONTAP LUNs.

To ensure that multipathing is configured correctly for your host, verify that the `/etc/multipath.conf` file is defined and that you have the NetApp recommended settings configured for your ONTAP LUNs.

Steps

1. Verify that the `/etc/multipath.conf` file exits. If the file doesn't exist, create an empty, zero-byte file:

```
touch /etc/multipath.conf
```

2. The first time the `multipath.conf` file is created, you might need to enable and start the multipath services to load the recommended settings:

```
systemctl enable multipathd
```

```
systemctl start multipathd
```

3. Each time you boot the host, the empty `/etc/multipath.conf` zero-byte file automatically loads the NetApp recommended host multipath parameters as the default settings. You shouldn't need to make changes to the `/etc/multipath.conf` file for your host because the operating system is compiled with the multipath parameters that recognize and manage ONTAP LUNs correctly.

   The following table shows the Linux OS native compiled multipath parameter settings for ONTAP LUNs.

Show parameter settings

Parameter	Setting
detect_prio	yes
dev_loss_tmo	"infinity"
failback	immediate
fast_io_fail_tmo	5
features	"2 pg_init_retries 50"
flush_on_last_del	"yes"
hardware_handler	"0"
no_path_retry	queue
path_checker	"tur"
path_grouping_policy	"group_by_prio"
path_selector	"service-time 0"
polling_interval	5
prio	"ontap"
product	LUN
retain_attached_hw_handler	yes
rr_weight	"uniform"
user_friendly_names	no
vendor	NETAPP

4. Verify the parameter settings and path status for your ONTAP LUNs:

```
multipath -ll
```

The default multipath parameters support ASA, AFF, and FAS configurations. In these configurations, a single ONTAP LUN shouldn’t require more than four paths. Having more than four paths can cause problems during a storage failure.

The following example outputs show the correct parameter settings and path status for ONTAP LUNs in an ASA, AFF, or FAS configuration.

ASA configuration

An ASA configuration optimizes all paths to a given LUN, keeping them active. This improves performance by serving I/O operations through all paths at the same time.

Show example

```
# multipath -ll
3600a0980383145595533f524d6c652f62 dm-24 NETAPP,LUN C-Mode
size=10G features='3 queue_if_no_path pg_init_retries 50'
hwhandler='1 alua' wp=rw
`-+- policy='service-time 0' prio=50 status=active
  |- 11:0:1:13 sdm  8:192  active ready running
  |- 11:0:3:13 sdah 66:16  active ready running
  |- 12:0:1:13 sdbc 67:96  active ready running
  `- 12:0:3:13 sdbx 68:176 active ready running
```

AFF or FAS configuration

An AFF or FAS configuration should have two groups of paths with higher and lower priorities. Higher priority Active/Optimized paths are served by the controller where the aggregate is located. Lower priority paths are active but non-optimized because they are served by a different controller. Non-optimized paths are only used when optimized paths aren’t available.

The following example displays the output for an ONTAP LUN with two Active/Optimized paths and two Active/Non-Optimized paths:

Show example

```
# multipath -ll
3600a0980383148373524536946b542f4a dm-0 NETAPP,LUN C-Mode
size=160G features='3 queue_if_no_path pg_init_retries 50'
hwhandler='1 alua' wp=rw
|-+- policy='service-time 0' prio=50 status=active
| |- 14:0:3:0 sdbk 67:224 active ready running
| `- 15:0:2:0 sdbl 67:240 active ready running
`-+- policy='service-time 0' prio=10 status=enabled
  |- 14:0:0:0 sda  8:0    active ready running
  `- 15:0:1:0 sdv  65:80  active ready running
```

Step 3: Confirm the iSCSI configuration for your host

Ensure that iSCSI is configured correctly for your host.

About this task

You perform the following steps on the iSCSI host.

Steps

1. Verify that the iSCSI initiator package (open-iscsi) is installed:

   ```
   $apt list |grep open-iscsi
   ```

   You should see an output similar to the following example:

   ```
   open-iscsi/noble-updates,noble-updates,now 2.1.9-3ubuntu5.4 amd64
   ```

2. Verify the iSCSI initiator node name, which is located in the `/etc/iscsi/initiatorname.iscsi` file:

   ```
   InitiatorName=iqn.YYYY-MM.com.<vendor>:<host_name>
   ```

3. Configure the iSCSI session timeout parameter located in the `/etc/iscsi/iscsid.conf` file:

   ```
   node.session.timeo.replacement_timeout = 5
   ```

   The iSCSI `replacement_timeout` parameter controls how long the iSCSI layer should wait for a timed-out path or session to reestablish itself before failing any commands on it. You should set the value of `replacement_timeout` to 5 in the iSCSI configuration file.

4. Enable the iSCSI service:

   ```
   $systemctl enable iscsid
   ```

5. Start the iSCSI service:

   ```
   $systemctl start iscsid
   ```

6. Verify that the iSCSI service is running:

   ```
   $systemctl status iscsid
   ```

Show example

```
●iscsid.service - iSCSI initiator daemon (iscsid)
     Loaded: loaded (/usr/lib/systemd/system/iscsid.service;
enabled; preset: disabled)
     Active: active (running) since Mon 2026-01-12 12:53:18 IST; 2
days ago
TriggeredBy: ● iscsid.socket
       Docs: man:iscsid(8)
   Main PID: 1127419 (iscsid)
      Tasks: 2 (limit: 76557)
     Memory: 4.3M (peak: 8.8M)
        CPU: 1.657s
     CGroup: /system.slice/iscsid.service
             ├─1127418 /usr/sbin/iscsid
             └─1127419 /usr/sbin/iscsid
```

7. Discover the iSCSI targets:

```
$iscsiadm --mode discovery --op update --type sendtargets --portal
<target_IP>
```

show example

```
iscsiadm --mode discovery --op update --type sendtargets --portal
192.168.100.197
192.168.100.197:3260,1046 iqn.1992-
08.com.netapp:sn.7cd154a7d35411f0a25ed039eaa95f59:vs.8
192.168.200.199:3260,1049 iqn.1992-
08.com.netapp:sn.7cd154a7d35411f0a25ed039eaa95f59:vs.8
192.168.100.199:3260,1048 iqn.1992-
08.com.netapp:sn.7cd154a7d35411f0a25ed039eaa95f59:vs.8
192.168.200.197:3260,1047 iqn.1992-
08.com.netapp:sn.7cd154a7d35411f0a25ed039eaa95f59:vs.8
```

8. Log in to the targets:

```
$iscsiadm --mode node -l all
```

9. Set iSCSI to log in automatically when the host boots:

```
$iscsiadm --mode node -T <target_name> -p <ip:port> -o update -n
node.startup -v automatic
```

You should see an output similar to the following example:

```
iscsiadm --mode node -T iqn.1992-
08.com.netapp:sn.7cd154a7d35411f0a25ed039eaa95f59:vs.8 -p
192.168.100.197:3260 -o update -n node.startup -v automatic
```

10. Verify the iSCSI sessions:

```
$iscsiadm --mode session
```

Show example

```
iscsiadm --mode session
tcp: [1] 192.168.200.197:3260,1047 iqn.1992-
08.com.netapp:sn.7cd154a7d35411f0a25ed039eaa95f59:vs.8 (non-flash)
tcp: [2] 192.168.100.197:3260,1046 iqn.1992-
08.com.netapp:sn.7cd154a7d35411f0a25ed039eaa95f59:vs.8 (non-flash)
tcp: [3] 192.168.100.199:3260,1048 iqn.1992-
08.com.netapp:sn.7cd154a7d35411f0a25ed039eaa95f59:vs.8 (non-flash)
tcp: [4] 192.168.200.199:3260,1049 iqn.1992-
08.com.netapp:sn.7cd154a7d35411f0a25ed039eaa95f59:vs.8 (non-flash)
```

Step 4: Optionally, exclude a device from multipathing

If required, you can exclude a device from multipathing by adding the WWID for the unwanted device to the "blacklist" stanza for the `multipath.conf` file.

Steps

1. Determine the WWID:

```
/lib/udev/scsi_id -gud /dev/sda
```

"sda" is the local SCSI disk that you want to add to the blacklist.

An example WWID is `360030057024d0730239134810c0cb833`.

2. Add the WWID to the "blacklist" stanza:

```
blacklist {
         wwid   360030057024d0730239134810c0cb833
        devnode "^(ram|raw|loop|fd|md|dm-|sr|scd|st)[0-9]*"
        devnode "^hd[a-z]"
        devnode "^cciss.*"
}
```

Step 5: Customize multipath parameters for ONTAP LUNs

If your host is connected to LUNs from other vendors and any of the multipath parameter settings are overridden, you need to correct them by adding stanzas later in the `multipath.conf` file that apply specifically to ONTAP LUNs. If you don't do this, the ONTAP LUNs might not work as expected.

Check your `/etc/multipath.conf` file, especially in the defaults section, for settings that might be overriding the default settings for multipath parameters.

> ⚠ You shouldn't override the recommended parameter settings for ONTAP LUNs. These settings are required for optimal performance of your host configuration. Contact NetApp support, your OS vendor, or both for more information.

The following example shows how to correct an overridden default. In this example, the `multipath.conf` file defines values for `path_checker` and `no_path_retry` that aren't compatible with ONTAP LUNs, and you can't remove these parameters because ONTAP storage arrays are still attached to the host. Instead, you correct the values for `path_checker` and `no_path_retry` by adding a device stanza to the `multipath.conf` file that applies specifically to the ONTAP LUNs.

Show example

```
defaults {
   path_checker       readsector0
   no_path_retry      fail
}

devices {
   device {
      vendor           "NETAPP"
      product          "LUN"
      no_path_retry    queue
      path_checker     tur
   }
}
```

Step 6: Review the known issues

There are no known issues.

What's next?

- Learn about Ubuntu Linux Virtualization (KVM)

  Ubuntu Linux can serve as a KVM host. This enables you to run multiple virtual machines on a single physical server using the Linux Kernel-based Virtual Machine (KVM) technology. The KVM host doesn't require explicit host configuration settings for ONTAP LUNs.

Veritas

Configure Veritas Infoscale 9 for FC, FCoE, and iSCSI with ONTAP storage

The Linux Host Utilities software provides management and diagnostic tools for Linux hosts that are connected to ONTAP storage. Use the Linux Host Utilities with Veritas Infoscale 9 for Oracle Linux (Red Hat Compatible Kernel based), Red Hat Enterprise Linux (RHEL), and SUSE Linux Enterprise Server hosts to support managing FC, FCoE, and iSCSI protocol operations with ONTAP LUNs.

Step 1: Optionally, enable SAN booting

You can configure your host to use SAN booting to simplify deployment and improve scalability.

Before you begin

- Use the Interoperability Matrix Tool to verify that your Linux OS, host bus adapter (HBA), HBA firmware, HBA boot BIOS, and ONTAP version support SAN booting.
- Refer to the Veritas Support Portal (Product Matrix, Platform Lookup, and HCL Matrix) to verify the SAN boot configuration support and known issues.

Steps

1. Create a SAN boot LUN and map it to the host.
2. Enable SAN booting in the server BIOS for the ports to which the SAN boot LUN is mapped.

   For information on how to enable the HBA BIOS, see your vendor-specific documentation.

3. Verify that the configuration was successful by rebooting the host and verifying that the OS is up and running.

Step 2: Install the Linux Host Utilities

NetApp strongly recommends installing the Linux Host Utilities to support ONTAP LUN management and assist technical support with gathering configuration data.

Installing the Linux Host Utilities doesn't change any host timeout settings on your Linux host.

Step 3: Confirm the Veritas Dynamic Multipathing configuration for your host

Use Veritas Dynamic Multipathing (VxDMP) with Veritas Infoscale 9 to manage ONTAP LUNs.

To ensure that VxDMP is set up correctly for your host, you need to verify the VxDMP configuration and check the Array Support Library (ASL) and the Array Policy Module (APM) configuration. The ASL and APM

packages for NetApp storage systems are installed during the Veritas software installation.

> For heterogenous multipath environments, including Veritas Infoscale, Linux Native Device Mapper, and LVM volume manager, refer to the Veritas Product Administration documentation for the configuration settings.

Before you begin

Ensure your configuration meets the system requirements. See the Interoperability Matrix Tool and the Veritas HCL Matrix.

Steps

1. Verify the ONTAP target array is attached to the VxDMP multipath:

```
vxdmpadm
```

Show example

```
#vxdmpadm listenclosure
ENCLR_NAME        ENCLR_TYPE     ENCLR_SNO      STATUS
ARRAY_TYPE     LUN_COUNT    FIRMWARE
================================================================================
================================
info_asa0         Info_ASA       81KDT+YTg35P         CONNECTED
ALUA       20          9161
infoscal1         Infoscal       81Ocq?Z7hPzC         CONNECTED
ALUA       23          9181
# vxdmpadm getdmpnode
NAME       STATE    ENCLR-TYPE  PATHS  ENBL  DSBL ENCLR-NAME
================================================================================
infoscal1_22         ENABLED      Infoscal     4       4     0
infoscal1
```

2. Check the configuration of the ASL and APM packages. NetApp recommends that you use the latest supported packages listed on the Veritas support portal.

Show example ASL and APM configuration

```
#  vxdmpadm list dmpnode dmpnodename=infoscal1_22 | grep asl
asl             = libvxnetapp.so

# vxddladm listversion |grep libvxnetapp.so
libvxnetapp.so                  vm-8.0.0-rev-1   8.0

# rpm -qa |grep VRTSaslapm
VRTSaslapm-9.0.3-RHEL9.x86_64
#vxddladm listsupport libname=libvxnetapp.so
ATTR_NAME   ATTR_VALUE
=========================================
LIBNAME     libvxnetapp.so
VID         NETAPP
PID         All
ARRAY_TYPE  ALUA, A/A
```

3. For optimum system configuration in storage failover operations, verify that you have the following Veritas VxDMP tunables:

Parameter	Setting
dmp_lun_retry_timeout	60
dmp_path_age	120
dmp_restore_interval	60

4. Set the DMP tunables to online:

```
# vxdmpadm settune dmp_tunable=value
```

5. Verify that the tunables settings are correct:

```
# vxdmpadm gettune
```

The following example shows the effective VxDMP tunables on a SAN host.

Show example

```
# vxdmpadm gettune

Tunable                    Current Value    Default Value
dmp_cache_open                        on               on
dmp_daemon_count                      10               10
dmp_delayq_interval                   15               15
dmp_display_alua_states               on               on
dmp_fast_recovery                     on               on
dmp_health_time                       60               60
dmp_iostats_state                enabled          enabled
dmp_log_level                          1                1
dmp_low_impact_probe                  on               on
dmp_lun_retry_timeout                 60               30
dmp_path_age                         120              300
dmp_pathswitch_blks_shift              9                9
dmp_probe_idle_lun                    on               on
dmp_probe_threshold                    5                5
dmp_restore_cycles                    10               10
dmp_restore_interval                  60              300
dmp_restore_policy         check_disabled   check_disabled
dmp_restore_state                enabled          enabled
dmp_retry_count                        5                5
dmp_scsi_timeout                      20               20
dmp_sfg_threshold                      1                1
dmp_stat_interval                      1                1
dmp_monitor_ownership                 on               on
dmp_monitor_fabric                    on               on
dmp_native_support                   off              off
```

6. Configure the protocol timeout values:

FC/FCoE

Use the default timeout values for FC and FCoE.

iSCSI

Set the `replacement_timeout` parameter value to 120.

The iSCSI `replacement_timeout` parameter controls how long the iSCSI layer should wait for a timed-out path or session to reestablish itself before failing any commands on it. NetApp recommends setting the value of `replacement_timeout` to 120 in the iSCSI configuration file.

```
# grep replacement_timeout /etc/iscsi/iscsid.conf
node.session.timeo.replacement_timeout = 120
```

7. Verify the parameter settings and path status for your ONTAP LUNs:

   In AFF, FAS, or ASA configurations, a single ONTAP LUN should not require more than four paths. More than four paths can cause problems during a storage failure.

   The following examples show the correct parameter settings and path status for ONTAP LUNs in an ASA, AFF, or FAS configuration.

ASA configuration

An ASA configuration optimizes all paths to a given LUN, keeping them active. This improves performance by serving I/O operations through all paths at the same time.

Show example

```
# vxdmpadm getsubpaths dmpnodename=infoscal1_21
NAME   STATE[A]    PATH-TYPE[M] CTLR-NAME  ENCLR-TYPE  ENCLR-
NAME   ATTRS  PRIORITY
=============================================================
==================
sdby   ENABLED(A)  Active/Optimized  c1    Infoscal    infoscal1
-      -
sddx   ENABLED(A)  Active/Optimized  c2    Infoscal    infoscal1
-      -
sdfe   ENABLED(A)  Active/Optimized  c1    Infoscal    infoscal1
-      -
sdfo   ENABLED(A)  Active/Optimized  c2    Infoscal    infoscal1
-      -
```

AFF or FAS configuration

An AFF or FAS configuration should have two groups of paths with higher and lower priorities. Higher priority Active/Optimized paths are served by the controller where the aggregate is located. Lower priority paths are active but non-optimized because they are served by a different controller. Non-optimized paths are only used when optimized paths aren’t available.

The following example displays the output for an ONTAP LUN with two Active/Optimized paths and two Active/Non-Optimized paths:

Show example

```
# vxdmpadm getsubpaths dmpnodename-sfrac0_47
NAME   STATE[A]    PATH-TYPE[M]    CTLR-NAME    ENCLR-TYPE   ENCLR-
NAME   ATTRS   PRIORITY
=============================================================
==================
sdas   ENABLED     Active/Non-Optimized c13    SFRAC        sfrac0
-       -
sdb    ENABLED(A)  Active/Optimized     c14    SFRAC        sfrac0
-       -
sdcj   ENABLED(A)  Active/Optimized     c14    SFRAC        sfrac0
-       -
sdea   ENABLED     Active/Non-Optimized c14    SFRAC        sfrac0
-       -
```

Step 4: Known issues

There are no known issues.

What's next?

- Learn about using the Linux Host Utilities tool.

Configure Veritas Infoscale 8 for FC, FCoE, and iSCSI with ONTAP storage

The Linux Host Utilities software provides management and diagnostic tools for Linux hosts that are connected to ONTAP storage. Use the Linux Host Utilities with Veritas Infoscale 8 for Oracle Linux (Red Hat Compatible Kernel based), Red Hat Enterprise Linux (RHEL), and SUSE Linux Enterprise Server hosts to support managing FC, FCoE, and iSCSI protocol operations with ONTAP LUNs.

Step 1: Optionally, enable SAN booting

You can configure your host to use SAN booting to simplify deployment and improve scalability.

Before you begin

- Use the Interoperability Matrix Tool to verify that your Linux OS, host bus adapter (HBA), HBA firmware, HBA boot BIOS, and ONTAP version support SAN booting.
- Refer to the Veritas Support Portal (Product Matrix, Platform Lookup, and HCL Matrix) to verify the SAN boot configuration support and known issues.

Steps

1. Create a SAN boot LUN and map it to the host.
2. Enable SAN booting in the server BIOS for the ports to which the SAN boot LUN is mapped.

For information on how to enable the HBA BIOS, see your vendor-specific documentation.

3. Verify that the configuration was successful by rebooting the host and verifying that the OS is up and running.

Step 2: Install the Linux Host Utilities

NetApp strongly recommends installing the Linux Host Utilities to support ONTAP LUN management and assist technical support with gathering configuration data.

Installing the Linux Host Utilities doesn't change any host timeout settings on your Linux host.

Step 3: Confirm the Veritas Dynamic Multipathing configuration for your host

Use Veritas Dynamic Multipathing (VxDMP) with Veritas Infoscale 8 to manage ONTAP LUNs.

To ensure that VxDMP is set up correctly for your host, you need to verify the VxDMP configuration and check the Array Support Library (ASL) and the Array Policy Module (APM) configuration. The ASL and APM packages for NetApp storage systems are installed during the Veritas software installation.

For heterogenous multipath environments, including Veritas Infoscale, Linux Native Device Mapper, and LVM volume manager, refer to the Veritas Product Administration documentation for the configuration settings.

Before you begin

Ensure that your configuration meets the system requirements. See the Interoperability Matrix Tool and the Veritas HCL Matrix.

Steps

1. Verify the ONTAP target array is attached to the VxDMP multipath:

```
vxdmpadm
```

Show example

```
# vxdmpadm listenclosure
ENCLR_NAME    ENCLR_TYPE  ENCLR_SNO      STATUS       ARRAY_TYPE
LUN_COUNT     FIRMWARE
=============================================================================
=================
sfrac0        SFRAC       804Xw$PqE52h   CONNECTED    ALUA           43
9800
# vxdmpadm getdmpnode
NAME        STATE     ENCLR-TYPE   PATHS   ENBL   DSBL ENCLR-NAME
=============================================================================
sfrac0_47   ENABLED   SFRAC        4       4      0    sfrac0
```

2. Check the configuration of the ASL and APM packages. NetApp recommends that you use the latest supported packages listed on the Veritas support portal.

   Show example ASL and APM configuration

```
# vxdmpadm list dmpnode dmpnodename=sfrac0_47 | grep asl
asl        = libvxnetapp.so
# vxddladm listversion |grep libvxnetapp.so
libvxnetapp.so                  vm-8.0.0-rev-1   8.0

# rpm -qa |grep VRTSaslapm
VRTSaslapm-x.x.x.0000-RHEL8.X86_64
vxddladm listsupport libname=libvxnetapp.so
ATTR_NAME   ATTR_VALUE
=======================================
LIBNAME     libvxnetapp.so
VID         NETAPP
PID         All
ARRAY_TYPE  ALUA, A/A
```

3. For optimum system configuration in storage failover operations, verify that you have the following Veritas VxDMP tunables:

Parameter	Setting
dmp_lun_retry_timeout	60
dmp_path_age	120
dmp_restore_interval	60

4. Set the DMP tunables to online:

```
# vxdmpadm settune dmp_tunable=value
```

5. Verify that the tunables settings are correct:

```
# vxdmpadm gettune
```

   The following example shows the effective VxDMP tunables on a SAN host.

Show example

```
# vxdmpadm gettune

Tunable                     Current Value    Default Value
dmp_cache_open                         on               on
dmp_daemon_count                       10               10
dmp_delayq_interval                    15               15
dmp_display_alua_states                on               on
dmp_fast_recovery                      on               on
dmp_health_time                        60               60
dmp_iostats_state                 enabled          enabled
dmp_log_level                           1                1
dmp_low_impact_probe                   on               on
dmp_lun_retry_timeout                  60               30
dmp_path_age                          120              300
dmp_pathswitch_blks_shift               9                9
dmp_probe_idle_lun                     on               on
dmp_probe_threshold                     5                5
dmp_restore_cycles                     10               10
dmp_restore_interval                   60              300
dmp_restore_policy          check_disabled   check_disabled
dmp_restore_state                 enabled          enabled
dmp_retry_count                         5                5
dmp_scsi_timeout                       20               20
dmp_sfg_threshold                       1                1
dmp_stat_interval                       1                1
dmp_monitor_ownership                  on               on
dmp_monitor_fabric                     on               on
dmp_native_support                    off              off
```

6. Configure the protocol timeout values:

FC/FCoE

Use the default timeout values for FC and FCoE.

iSCSI

Set the `replacement_timeout` parameter value to 120.

The iSCSI `replacement_timeout` parameter controls how long the iSCSI layer should wait for a timed-out path or session to reestablish itself before failing any commands on it. NetApp recommends setting the value of `replacement_timeout` to 120 in the iSCSI configuration file.

```
# grep replacement_timeout /etc/iscsi/iscsid.conf
node.session.timeo.replacement_timeout = 120
```

7. Set the host "udev rport" values for RHEL 8 and 9 series hosts to support the Veritas Infoscale environment in storage failover scenarios.

   Configure "udev rport" values by creating the file `/etc/udev/rules.d/40-rport.rules` with the following file content:

```
# cat /etc/udev/rules.d/40-rport.rules
KERNEL=="rport-*", SUBSYSTEM=="fc_remote_ports", ACTION=="add",
RUN+=/bin/sh -c 'echo 20 >
/sys/class/fc_remote_ports/%k/fast_io_fail_tmo;echo 864000
>/sys/class/fc_remote_ports/%k/dev_loss_tmo'"
```

   Refer to the standard Veritas Infoscale product documentation for all other settings specific to Veritas.

8. Verify the parameter settings and path status for your ONTAP LUNs:

   In AFF, FAS, or ASA configurations, a single ONTAP LUN should not require more than four paths. More than four paths can cause problems during a storage failure.

   The following examples show the correct parameter settings and path status for ONTAP LUNs in an ASA, AFF, or FAS configuration.

ASA configuration

An ASA configuration optimizes all paths to a given LUN, keeping them active. This improves performance by serving I/O operations through all paths at the same time.

Show example

```
# vxdmpadm getsubpaths dmpnodename-sfrac0_47
NAME  STATE[A]   PATH-TYPE[M]   CTLR-NAME   ENCLR-TYPE  ENCLR-
NAME  ATTRS  PRIORITY
=============================================================
==================
sdas  ENABLED (A)    Active/Optimized c13   SFRAC       sfrac0
-      -
sdb   ENABLED(A) Active/Optimized     c14   SFRAC       sfrac0
-      -
sdcj  ENABLED(A)  Active/Optimized     c14   SFRAC       sfrac0
-      -
sdea  ENABLED (A)    Active/Optimized c14   SFRAC       sfrac0
-
```

AFF or FAS configuration

An AFF or FAS configuration should have two groups of paths with higher and lower priorities. Higher priority Active/Optimized paths are served by the controller where the aggregate is located. Lower priority paths are active but non-optimized because they are served by a different controller. Non-optimized paths are only used when optimized paths aren’t available.

The following example displays the output for an ONTAP LUN with two Active/Optimized paths and two Active/Non-Optimized paths:

Show example

```
# vxdmpadm getsubpaths dmpnodename-sfrac0_47
NAME   STATE[A]    PATH-TYPE[M]    CTLR-NAME    ENCLR-TYPE   ENCLR-
NAME   ATTRS   PRIORITY
=============================================================
==================
sdas   ENABLED     Active/Non-Optimized c13    SFRAC        sfrac0
-       -
sdb    ENABLED(A)  Active/Optimized     c14    SFRAC        sfrac0
-       -
sdcj   ENABLED(A)  Active/Optimized     c14    SFRAC        sfrac0
-       -
sdea   ENABLED     Active/Non-Optimized c14    SFRAC        sfrac0
-       -
```

Step 4: Known issues

There are no known issues.

What's next?

- Learn about using the Linux Host Utilities tool.

Configure Veritas Infoscale 7 for FC, FCoE, and iSCSI with ONTAP storage

The Linux Host Utilities software provides management and diagnostic tools for Linux hosts that are connected to ONTAP storage. Use the Linux Host Utilities with Veritas Infoscale 7 for Oracle Linux (Red Hat Compatible Kernel based), Red Hat Enterprise Linux (RHEL), and SUSE Linux Enterprise Server hosts to support managing FC, FCoE, and iSCSI protocol operations with ONTAP LUNs.

Step 1: Optionally, enable SAN booting

You can configure your host to use SAN booting to simplify deployment and improve scalability.

Before you begin

- Use the Interoperability Matrix Tool to verify that your Linux OS, host bus adapter (HBA), HBA firmware, HBA boot BIOS, and ONTAP version support SAN booting.
- Refer to the Veritas Support Portal (Product Matrix, Platform Lookup, and HCL Matrix) to verify the SAN boot configuration support and known issues.

Steps

1. Create a SAN boot LUN and map it to the host.
2. Enable SAN booting in the server BIOS for the ports to which the SAN boot LUN is mapped.

For information on how to enable the HBA BIOS, see your vendor-specific documentation.

3. Verify that the configuration was successful by rebooting the host and verifying that the OS is up and running.

Step 2: Install the Linux Host Utilities

NetApp strongly recommends installing the Linux Host Utilities to support ONTAP LUN management and assist technical support with gathering configuration data.

Installing the Linux Host Utilities doesn't change any host timeout settings on your Linux host.

Step 3: Confirm the Veritas Dynamic Multipathing configuration for your host

Use Veritas Dynamic Multipathing (VxDMP) with Veritas Infoscale 7 to manage ONTAP LUNs.

To ensure that VxDMP is set up correctly for your host, you need to verify the VxDMP configuration and check the Array Support Library (ASL) and the Array Policy Module (APM) configuration. The ASL and APM packages for NetApp storage systems are installed during the Veritas software installation.

For heterogenous multipath environments, including Veritas Infoscale, Linux Native Device Mapper, and LVM volume manager, refer to the Veritas Product Administration documentation for the configuration settings.

Before you begin

Ensure that your configuration meets the system requirements. See the Interoperability Matrix Tool and the Veritas HCL Matrix.

Steps

1. Verify the ONTAP target array is attached to the VxDMP multipath:

```
vxdmpadm
```

Show example

```
# vxdmpadm listenclosure
ENCLR_NAME    ENCLR_TYPE  ENCLR_SNO      STATUS       ARRAY_TYPE
LUN_COUNT     FIRMWARE
================================================================================
=================
sfrac0        SFRAC       804Xw$PqE52h   CONNECTED    ALUA           43
9800
# vxdmpadm getdmpnode
NAME        STATE     ENCLR-TYPE   PATHS   ENBL   DSBL ENCLR-NAME
================================================================================
sfrac0_47   ENABLED   SFRAC        4       4      0    sfrac0
```

2. Check the configuration of the ASL and APM packages. NetApp recommends that you use the latest supported packages listed on the Veritas support portal.

   Show example ASL and APM configuration

```
# vxdmpadm list dmpnode dmpnodename=sfrac0_47 | grep asl
asl        = libvxnetapp.so
# vxddladm listversion |grep libvxnetapp.so
libvxnetapp.so                  vm-8.0.0-rev-1   8.0

# rpm -qa |grep VRTSaslapm
VRTSaslapm-x.x.x.0000-RHEL8.X86_64
vxddladm listsupport libname=libvxnetapp.so
ATTR_NAME   ATTR_VALUE
=======================================
LIBNAME     libvxnetapp.so
VID         NETAPP
PID         All
ARRAY_TYPE  ALUA, A/A
```

3. For optimum system configuration in storage failover operations, verify that you have the following Veritas VxDMP tunables:

Parameter	Setting
dmp_lun_retry_timeout	60
dmp_path_age	120
dmp_restore_interval	60

4. Set the DMP tunables to online:

```
# vxdmpadm settune dmp_tunable=value
```

5. Verify that the tunables settings are correct:

```
# vxdmpadm gettune
```

   The following example shows the effective VxDMP tunables on a SAN host.

Show example

```
# vxdmpadm gettune

Tunable                     Current Value    Default Value
dmp_cache_open                         on               on
dmp_daemon_count                       10               10
dmp_delayq_interval                    15               15
dmp_display_alua_states                on               on
dmp_fast_recovery                      on               on
dmp_health_time                        60               60
dmp_iostats_state                 enabled          enabled
dmp_log_level                           1                1
dmp_low_impact_probe                   on               on
dmp_lun_retry_timeout                  60               30
dmp_path_age                          120              300
dmp_pathswitch_blks_shift               9                9
dmp_probe_idle_lun                     on               on
dmp_probe_threshold                     5                5
dmp_restore_cycles                     10               10
dmp_restore_interval                   60              300
dmp_restore_policy          check_disabled   check_disabled
dmp_restore_state                 enabled          enabled
dmp_retry_count                         5                5
dmp_scsi_timeout                       20               20
dmp_sfg_threshold                       1                1
dmp_stat_interval                       1                1
dmp_monitor_ownership                  on               on
dmp_monitor_fabric                     on               on
dmp_native_support                    off              off
```

6. Configure the protocol timeout values:

FC/FCoE

Use the default timeout values for FC and FCoE.

iSCSI

Set the `replacement_timeout` parameter value to 120.

The iSCSI `replacement_timeout` parameter controls how long the iSCSI layer should wait for a timed-out path or session to reestablish itself before failing any commands on it. NetApp recommends setting the value of `replacement_timeout` to 120 in the iSCSI configuration file.

```
# grep replacement_timeout /etc/iscsi/iscsid.conf
node.session.timeo.replacement_timeout = 120
```

7. Set the host "udev rport" values for RHEL 8 and 9 series hosts to support the Veritas Infoscale environment in storage failover scenarios.

   Configure "udev rport" values by creating the file `/etc/udev/rules.d/40-rport.rules` with the following file content:

```
# cat /etc/udev/rules.d/40-rport.rules
KERNEL=="rport-*", SUBSYSTEM=="fc_remote_ports", ACTION=="add",
RUN+=/bin/sh -c 'echo 20 >
/sys/class/fc_remote_ports/%k/fast_io_fail_tmo;echo 864000
>/sys/class/fc_remote_ports/%k/dev_loss_tmo'"
```

   Refer to the standard Veritas Infoscale product documentation for all other settings specific to Veritas.

8. Verify the parameter settings and path status for your ONTAP LUNs:

   In AFF, FAS, or ASA configurations, a single ONTAP LUN should not require more than four paths. More than four paths can cause problems during a storage failure.

   The following examples show the correct parameter settings and path status for ONTAP LUNs in an ASA, AFF, or FAS configuration.

ASA configuration

An ASA configuration optimizes all paths to a given LUN, keeping them active. This improves performance by serving I/O operations through all paths at the same time.

Show example

```
# vxdmpadm getsubpaths dmpnodename-sfrac0_47
NAME  STATE[A]   PATH-TYPE[M]   CTLR-NAME   ENCLR-TYPE  ENCLR-
NAME  ATTRS  PRIORITY
=============================================================
==================
sdas  ENABLED (A)    Active/Optimized c13   SFRAC       sfrac0
-      -
sdb   ENABLED(A) Active/Optimized     c14   SFRAC       sfrac0
-      -
sdcj  ENABLED(A)  Active/Optimized     c14   SFRAC       sfrac0
-      -
sdea  ENABLED (A)    Active/Optimized c14   SFRAC       sfrac0
-
```

AFF or FAS configuration

An AFF or FAS configuration should have two groups of paths with higher and lower priorities. Higher priority Active/Optimized paths are served by the controller where the aggregate is located. Lower priority paths are active but non-optimized because they are served by a different controller. Non-optimized paths are only used when optimized paths aren’t available.

The following example displays the output for an ONTAP LUN with two Active/Optimized paths and two Active/Non-Optimized paths:

Show example

```
# vxdmpadm getsubpaths dmpnodename-sfrac0_47
NAME   STATE[A]    PATH-TYPE[M]    CTLR-NAME    ENCLR-TYPE   ENCLR-
NAME   ATTRS   PRIORITY
=============================================================
==================
sdas   ENABLED     Active/Non-Optimized c13    SFRAC        sfrac0
-       -
sdb    ENABLED(A)  Active/Optimized     c14    SFRAC        sfrac0
-       -
sdcj   ENABLED(A)  Active/Optimized     c14    SFRAC        sfrac0
-       -
sdea   ENABLED     Active/Non-Optimized c14    SFRAC        sfrac0
-       -
```

Step 4: Known issues

There are no known issues.

What's next?

- Learn about using the Linux Host Utilities tool.

Configure Veritas Infoscale 6 for FC, FCoE, and iSCSI with ONTAP storage

The Linux Host Utilities software provides management and diagnostic tools for Linux hosts that are connected to ONTAP storage. Use the Linux Host Utilities with Veritas Infoscale 6 for Oracle Linux (Red Hat Compatible Kernel based), Red Hat Enterprise Linux (RHEL), and SUSE Linux Enterprise Server hosts to support managing FC, FCoE, and iSCSI protocol operations with ONTAP LUNs.

Step 1: Optionally, enable SAN booting

You can configure your host to use SAN booting to simplify deployment and improve scalability.

Before you begin

- Use the Interoperability Matrix Tool to verify that your Linux OS, host bus adapter (HBA), HBA firmware, HBA boot BIOS, and ONTAP version support SAN booting.
- Refer to the Veritas Support Portal (Product Matrix, Platform Lookup, and HCL Matrix) to verify the SAN boot configuration support and known issues.

Steps

1. Create a SAN boot LUN and map it to the host.
2. Enable SAN booting in the server BIOS for the ports to which the SAN boot LUN is mapped.

For information on how to enable the HBA BIOS, see your vendor-specific documentation.

3. Verify that the configuration was successful by rebooting the host and verifying that the OS is up and running.

Step 2: Install the Linux Host Utilities

NetApp strongly recommends installing the Linux Host Utilities to support ONTAP LUN management and assist technical support with gathering configuration data.

> Installing the Linux Host Utilities doesn't change any host timeout settings on your Linux host.

Step 3: Confirm the Veritas Dynamic Multipathing configuration for your host

Use Veritas Dynamic Multipathing (VxDMP) with Veritas Infoscale 6 to manage ONTAP LUNs.

To ensure that VxDMP is set up correctly for your host, you need to verify the VxDMP configuration and check the Array Support Library (ASL) and the Array Policy Module (APM) configuration. The ASL and APM packages for NetApp storage systems are installed during the Veritas software installation.

> For heterogenous multipath environments, including Veritas Infoscale, Linux Native Device Mapper, and LVM volume manager, refer to the Veritas Product Administration documentation for the configuration settings.

Before you begin

Ensure that your configuration meets the system requirements. See the Interoperability Matrix Tool and the Veritas HCL Matrix.

Steps

1. Verify the ONTAP target array is attached to the VxDMP multipath:

```
vxdmpadm
```

Show example

```
# vxdmpadm listenclosure
ENCLR_NAME    ENCLR_TYPE  ENCLR_SNO      STATUS       ARRAY_TYPE
LUN_COUNT     FIRMWARE
================================================================================
=================
sfrac0        SFRAC       804Xw$PqE52h   CONNECTED    ALUA           43
9800
# vxdmpadm getdmpnode
NAME        STATE     ENCLR-TYPE   PATHS   ENBL   DSBL ENCLR-NAME
================================================================================
sfrac0_47   ENABLED   SFRAC        4       4      0    sfrac0
```

2. Check the configuration of the ASL and APM packages. NetApp recommends that you use the latest supported packages listed on the Veritas support portal.

   Show example ASL and APM configuration

```
# vxdmpadm list dmpnode dmpnodename=sfrac0_47 | grep asl
asl        = libvxnetapp.so
# vxddladm listversion |grep libvxnetapp.so
libvxnetapp.so                  vm-8.0.0-rev-1   8.0

# rpm -qa |grep VRTSaslapm
VRTSaslapm-x.x.x.0000-RHEL8.X86_64
vxddladm listsupport libname=libvxnetapp.so
ATTR_NAME   ATTR_VALUE
=======================================
LIBNAME     libvxnetapp.so
VID         NETAPP
PID         All
ARRAY_TYPE  ALUA, A/A
```

3. For optimum system configuration in storage failover operations, verify that you have the following Veritas VxDMP tunables:

Parameter	Setting
dmp_lun_retry_timeout	60
dmp_path_age	120
dmp_restore_interval	60

4. Set the DMP tunables to online:

```
# vxdmpadm settune dmp_tunable=value
```

5. Verify that the tunables settings are correct:

```
# vxdmpadm gettune
```

   The following example shows the effective VxDMP tunables on a SAN host.

Show example

```
# vxdmpadm gettune

Tunable                     Current Value    Default Value
dmp_cache_open                         on               on
dmp_daemon_count                       10               10
dmp_delayq_interval                    15               15
dmp_display_alua_states                on               on
dmp_fast_recovery                      on               on
dmp_health_time                        60               60
dmp_iostats_state                 enabled          enabled
dmp_log_level                           1                1
dmp_low_impact_probe                   on               on
dmp_lun_retry_timeout                  60               30
dmp_path_age                          120              300
dmp_pathswitch_blks_shift               9                9
dmp_probe_idle_lun                     on               on
dmp_probe_threshold                     5                5
dmp_restore_cycles                     10               10
dmp_restore_interval                   60              300
dmp_restore_policy          check_disabled   check_disabled
dmp_restore_state                 enabled          enabled
dmp_retry_count                         5                5
dmp_scsi_timeout                       20               20
dmp_sfg_threshold                       1                1
dmp_stat_interval                       1                1
dmp_monitor_ownership                  on               on
dmp_monitor_fabric                     on               on
dmp_native_support                    off              off
```

6. Configure the protocol timeout values:

FC/FCoE

Use the default timeout values for FC and FCoE.

iSCSI

Set the `replacement_timeout` parameter value to 120.

The iSCSI `replacement_timeout` parameter controls how long the iSCSI layer should wait for a timed-out path or session to reestablish itself before failing any commands on it. NetApp recommends setting the value of `replacement_timeout` to 120 in the iSCSI configuration file.

```
# grep replacement_timeout /etc/iscsi/iscsid.conf
node.session.timeo.replacement_timeout = 120
```

7. Set the host "udev rport" values for RHEL 8 and 9 series hosts to support the Veritas Infoscale environment in storage failover scenarios.

   Configure "udev rport" values by creating the file `/etc/udev/rules.d/40-rport.rules` with the following file content:

```
# cat /etc/udev/rules.d/40-rport.rules
KERNEL=="rport-*", SUBSYSTEM=="fc_remote_ports", ACTION=="add",
RUN+=/bin/sh -c 'echo 20 >
/sys/class/fc_remote_ports/%k/fast_io_fail_tmo;echo 864000
>/sys/class/fc_remote_ports/%k/dev_loss_tmo'"
```

   Refer to the standard Veritas Infoscale product documentation for all other settings specific to Veritas.

8. Verify the parameter settings and path status for your ONTAP LUNs:

   In AFF, FAS, or ASA configurations, a single ONTAP LUN should not require more than four paths. More than four paths can cause problems during a storage failure.

   The following examples show the correct parameter settings and path status for ONTAP LUNs in an ASA, AFF, or FAS configuration.

ASA configuration

An ASA configuration optimizes all paths to a given LUN, keeping them active. This improves performance by serving I/O operations through all paths at the same time.

Show example

```
# vxdmpadm getsubpaths dmpnodename-sfrac0_47
NAME  STATE[A]   PATH-TYPE[M]   CTLR-NAME   ENCLR-TYPE  ENCLR-
NAME  ATTRS  PRIORITY
=============================================================
==================
sdas  ENABLED (A)    Active/Optimized c13   SFRAC       sfrac0
-      -
sdb   ENABLED(A) Active/Optimized     c14   SFRAC       sfrac0
-      -
sdcj  ENABLED(A)  Active/Optimized     c14   SFRAC       sfrac0
-      -
sdea  ENABLED (A)    Active/Optimized c14   SFRAC       sfrac0
-
```

AFF or FAS configuration

An AFF or FAS configuration should have two groups of paths with higher and lower priorities. Higher priority Active/Optimized paths are served by the controller where the aggregate is located. Lower priority paths are active but non-optimized because they are served by a different controller. Non-optimized paths are only used when optimized paths aren’t available.

The following example displays the output for an ONTAP LUN with two Active/Optimized paths and two Active/Non-Optimized paths:

Show example

```
# vxdmpadm getsubpaths dmpnodename-sfrac0_47
NAME   STATE[A]    PATH-TYPE[M]    CTLR-NAME    ENCLR-TYPE   ENCLR-
NAME   ATTRS   PRIORITY
=============================================================
==================
sdas   ENABLED     Active/Non-Optimized c13    SFRAC        sfrac0
-       -
sdb    ENABLED(A)  Active/Optimized     c14    SFRAC        sfrac0
-       -
sdcj   ENABLED(A)  Active/Optimized     c14    SFRAC        sfrac0
-       -
sdea   ENABLED     Active/Non-Optimized c14    SFRAC        sfrac0
-       -
```

Step 4: Known issues

There are no known issues.

What's next?

- Learn about using the Linux Host Utilities tool.

Windows

Configure Windows Server 2025 for FCP and iSCSI with ONTAP storage

The Windows Host Utilities are a set of software programs with documentation that enables you to connect Windows hosts to virtual disks (LUNs) on a NetApp SAN. When you install the Windows Host Utilities on a Windows Server 2025 host, you can use the Host Utilities to help you manage FCP and iSCSI protocol operations with ONTAP LUNs.
.

Step 1: Optionally, enable SAN booting

You can boot the Windows OS using a local boot or a SAN boot. NetApp recommends using a SAN boot to simplify deployment and improve scalability..

SAN boot

If you choose to use SAN booting, it must be supported by your configuration.

Before you begin

Use the Interoperability Matrix Tool to verify that your Windows OS, host bus adapter (HBA), HBA firmware, HBA boot BIOS, and ONTAP version support SAN booting.

Steps

1. Create a SAN boot LUN and map it to the host.
2. Enable SAN booting in the server BIOS for the ports to which the SAN boot LUN is mapped.

   For information on how to enable the HBA BIOS, see your vendor-specific documentation.

3. Verify that the configuration was successful by rebooting the host and verifying that the OS is up and running.

Local boot

Perform a local boot by installing the Windows OS on the local hard disk, for example, on an SSD, SATA, or RAID.

Step 2: Install Windows hotfixes

NetApp recommends installing the **latest cumulative update** available from the Microsoft Update Catalog on the host server.

Steps

1. Download the hotfixes from the Microsoft Update Catalog 2025.

   You need to contact Microsoft support for the hotfixes that aren't available for download from the Microsoft Update Catalog.

2. Follow the instructions provided by Microsoft to install the hotfixes.

   Many hotfixes require a Windows host reboot. You can wait to reboot the host until *after* you install or upgrade the Host Utilities.

Step 3: Install the Windows Host Utilities

The Windows Host Utilities are a set of software programs with documentation that enable you to connect host computers to virtual disks (LUNs) on a NetApp SAN. NetApp recommends downloading and installing the latest Windows Host Utilities to support ONTAP LUN management and help technical support collect configuration data.

For Windows Host Utilities configuration and installation information, see the Windows Host Utilities documentation and select the installation procedure for your Windows Host Utilities version.

Step 4: Confirm the multipath configuration for your host

Install the Microsoft Multipath I/O (MPIO) software and enable multipathing if your Windows host has more than one path to the storage system.

On a Windows system, the two main components in an MPIO solution are the device-specific module (DSM) and the Windows MPIO. MPIO presents one disk to the Windows OS for all paths and the DSM manages path failovers.

If you don't install the MPIO software, the Windows OS might see each path as a separate disk. This can lead to data corruption.

Windows XP or Windows Vista running in a Hyper-V virtual machine doesn't support MPIO.

Steps

1. Install the MPIO software and enable multipathing.
2. When you select MPIO on systems using FC, the Host Utilities installer sets the required timeout values for Emulex and QLogic FC HBAs.

   Emulex FC

   The timeout values for Emulex FC HBAs:

Property type	Property value
LinkTimeOut	1
NodeTimeOut	10

   QLogic FC

   The timeout values for QLogic FC HBAs:

Property type	Property value
LinkDownTimeOut	1
PortDownRetryCount	10

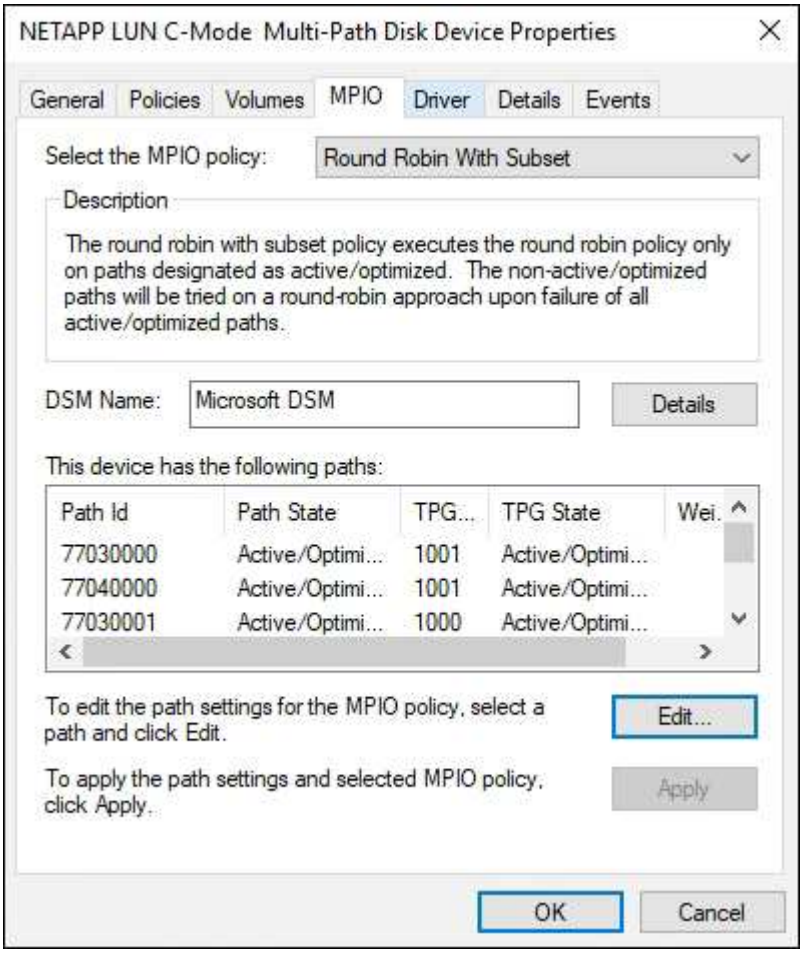
3. Verify the path status for your ONTAP LUNs:

   Depending on your SAN configuration, the host uses ASA, AFF, or FAS configurations to access ONTAP LUNs. These configurations shouldn't require more than four paths to access a single ONTAP LUN. More than four paths can cause problems during storage failure.

   The following example outputs show the correct settings for ONTAP LUNs for an ASA, AFF, or FAS configuration.

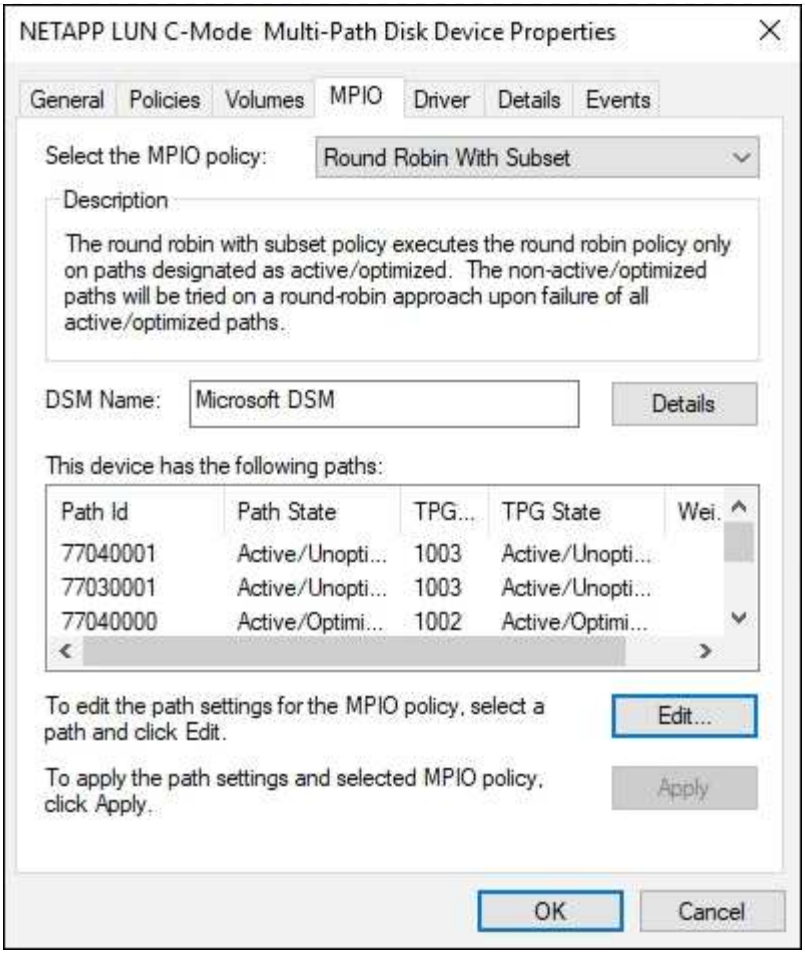
ASA configuration

An ASA configuration should have one group of Active/Optimized paths with single priorities. The controller services the paths and sends I/O on all active paths.



AFF or FAS configuration

An AFF or FAS configuration should have two groups of paths with different priorities. The paths with higher priorities are Active/Optimized and are serviced by the controller where the aggregate is located. The paths with lower priorities are serviced from a different controller. They are active but non-optimized and are only used when optimized paths aren't available.



Step 5: Review the Known issues

There are no known issues.

What's next?

Learn about the Windows Host Utilities configuration for ONTAP storage

Configure Windows Server 2022 for FCP and iSCSI with ONTAP storage

The Windows Host Utilities enable you to connect Windows hosts to virtual disks (LUNs) on a NetApp SAN. Install the Windows Host Utilities on a Windows Server 2022 host to help you manage FCP and iSCSI protocol operations with ONTAP LUNs.

Step 1: Optionally, enable SAN booting

You can boot the Windows OS using a local boot or a SAN boot. NetApp recommends using a SAN boot to simplify deployment and improve scalability..

SAN boot

If you choose to use SAN booting, it must be supported by your configuration.

Before you begin

Use the Interoperability Matrix Tool to verify that your Windows OS, host bus adapter (HBA), HBA firmware, HBA boot BIOS, and ONTAP version support SAN booting.

Steps

1. Create a SAN boot LUN and map it to the host.
2. Enable SAN booting in the server BIOS for the ports to which the SAN boot LUN is mapped.

   For information on how to enable the HBA BIOS, see your vendor-specific documentation.

3. Verify that the configuration was successful by rebooting the host and verifying that the OS is up and running.

Local boot

Perform a local boot by installing the Windows OS on the local hard disk, for example, on an SSD, SATA, or RAID.

Step 2: Install Windows hotfixes

NetApp recommends installing the **latest cumulative update** available from the Microsoft Update Catalog on the host server.

Steps

1. Download the hotfixes from the Microsoft Update Catalog 2022.

   You need to contact Microsoft support for the hotfixes that aren't available for download from the Microsoft Update Catalog.

2. Follow the instructions provided by Microsoft to install the hotfixes.

   Many hotfixes require a Windows host reboot. You can wait to reboot the host until *after* you install or upgrade the Host Utilities.

Step 3: Install the Windows Host Utilities

The Windows Host Utilities are a set of software programs with documentation that enable you to connect host computers to virtual disks (LUNs) on a NetApp SAN. NetApp recommends downloading and installing the latest Windows Host Utilities to support ONTAP LUN management and help technical support collect configuration data.

For Windows Host Utilities configuration and installation information, see the Windows Host Utilities documentation and select the installation procedure for your Windows Host Utilities version.

Step 4: Confirm the multipath configuration for your host

Install the Microsoft Multipath I/O (MPIO) software and enable multipathing if your Windows host has more than one path to the storage system.

On a Windows system, the two main components in an MPIO solution are the device-specific module (DSM) and the Windows MPIO. MPIO presents one disk to the Windows OS for all paths and the DSM manages path failovers.

If you don't install the MPIO software, the Windows OS might see each path as a separate disk. This can lead to data corruption.

Windows XP or Windows Vista running in a Hyper-V virtual machine doesn't support MPIO.

Steps

1. Install the MPIO software and enable multipathing.
2. When you select MPIO on systems using FC, the Host Utilities installer sets the required timeout values for Emulex and QLogic FC HBAs.

Emulex FC

The timeout values for Emulex FC HBAs:

Property type	Property value
LinkTimeOut	1
NodeTimeOut	10

QLogic FC

The timeout values for QLogic FC HBAs:

Property type	Property value
LinkDownTimeOut	1
PortDownRetryCount	10

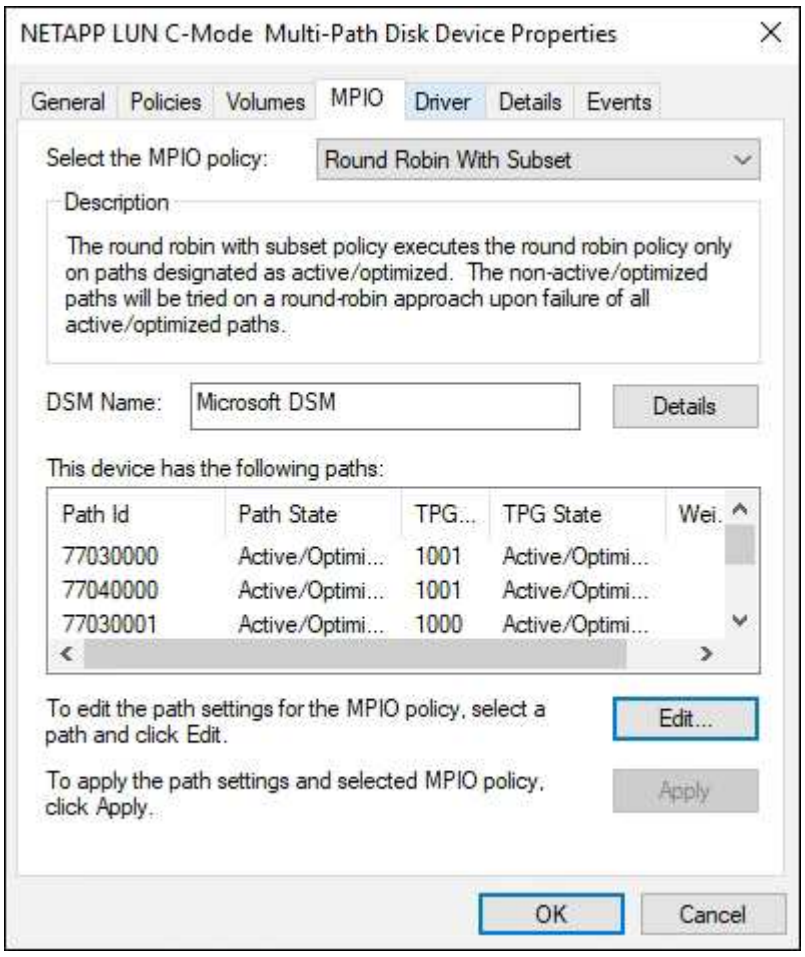
3. Verify the path status for your ONTAP LUNs:

   Depending on your SAN configuration, the host uses ASA, AFF, or FAS configurations to access ONTAP LUNs. These configurations shouldn't require more than four paths to access a single ONTAP LUN. More than four paths can cause problems during storage failure.

   The following example outputs show the correct settings for ONTAP LUNs for an ASA, AFF, or FAS configuration.

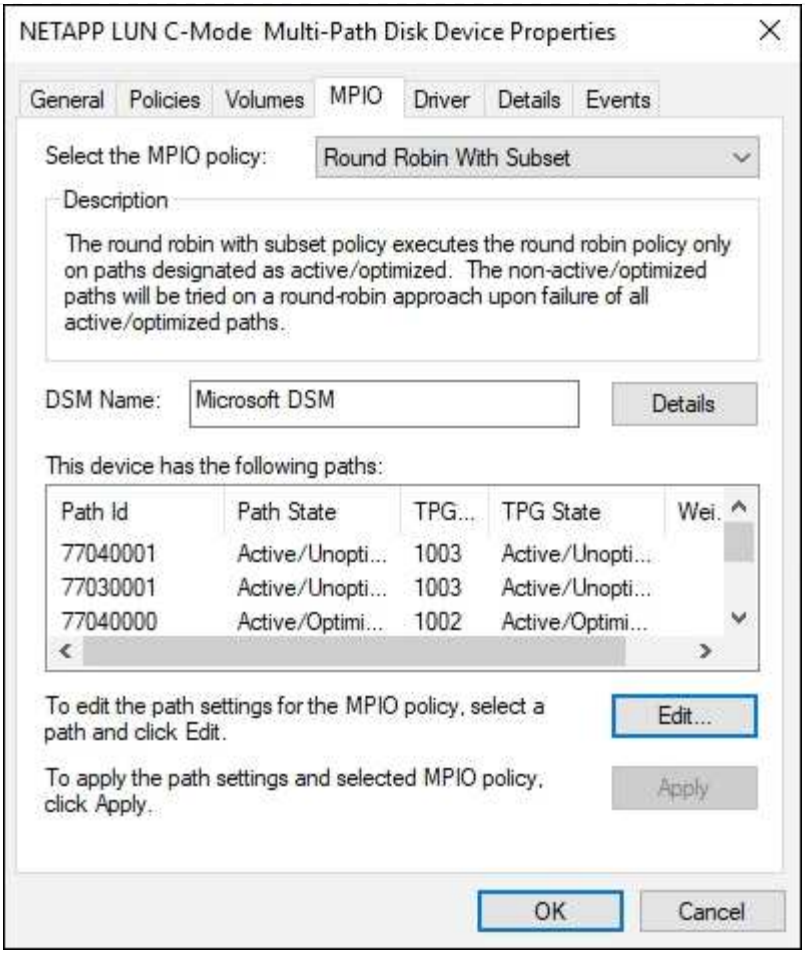
ASA configuration

An ASA configuration should have one group of Active/Optimized paths with single priorities. The controller services the paths and sends I/O on all active paths.



AFF or FAS configuration

An AFF or FAS configuration should have two groups of paths with different priorities. The paths with higher priorities are Active/Optimized and are serviced by the controller where the aggregate is located. The paths with lower priorities are serviced from a different controller. They are active but non-optimized and are only used when optimized paths aren't available.



Step 5: Review the Known issues

There are no known issues.

What's next?

Learn about the Windows Host Utilities configuration for ONTAP storage

Configure Windows Server 2019 for FCP and iSCSI with ONTAP storage

The Windows Host Utilities enable you to connect Windows hosts to virtual disks (LUNs) on a NetApp SAN. Install the Windows Host Utilities on a Windows Server 2019 host to help you manage FCP and iSCSI protocol operations with ONTAP LUNs.

Step 1: Optionally, enable SAN booting

You can boot the Windows OS using a local boot or a SAN boot. NetApp recommends using a SAN boot to simplify deployment and improve scalability..

SAN boot

If you choose to use SAN booting, it must be supported by your configuration.

Before you begin

Use the Interoperability Matrix Tool to verify that your Windows OS, host bus adapter (HBA), HBA firmware, HBA boot BIOS, and ONTAP version support SAN booting.

Steps

1. Create a SAN boot LUN and map it to the host.
2. Enable SAN booting in the server BIOS for the ports to which the SAN boot LUN is mapped.

   For information on how to enable the HBA BIOS, see your vendor-specific documentation.

3. Verify that the configuration was successful by rebooting the host and verifying that the OS is up and running.

Local boot

Perform a local boot by installing the Windows OS on the local hard disk, for example, on an SSD, SATA, or RAID.

Step 2: Install Windows hotfixes

NetApp recommends installing the **latest cumulative update** available from the Microsoft Update Catalog on the host server.

Steps

1. Download the hotfixes from the Microsoft Update Catalog 2019.

   You need to contact Microsoft support for the hotfixes that aren't available for download from the Microsoft Update Catalog.

2. Follow the instructions provided by Microsoft to install the hotfixes.

   Many hotfixes require a Windows host reboot. You can wait to reboot the host until *after* you install or upgrade the Host Utilities.

Step 3: Install the Windows Host Utilities

The Windows Host Utilities are a set of software programs with documentation that enable you to connect host computers to virtual disks (LUNs) on a NetApp SAN. NetApp recommends downloading and installing the latest Windows Host Utilities to support ONTAP LUN management and help technical support collect configuration data.

For Windows Host Utilities configuration and installation information, see the Windows Host Utilities documentation and select the installation procedure for your Windows Host Utilities version.

Step 4: Confirm the multipath configuration for your host

Install the Microsoft Multipath I/O (MPIO) software and enable multipathing if your Windows host has more than one path to the storage system.

On a Windows system, the two main components in an MPIO solution are the device-specific module (DSM) and the Windows MPIO. MPIO presents one disk to the Windows OS for all paths and the DSM manages path failovers.

If you don't install the MPIO software, the Windows OS might see each path as a separate disk. This can lead to data corruption.

Windows XP or Windows Vista running in a Hyper-V virtual machine doesn't support MPIO.

Steps

1. Install the MPIO software and enable multipathing.
2. When you select MPIO on systems using FC, the Host Utilities installer sets the required timeout values for Emulex and QLogic FC HBAs.

   Emulex FC

   The timeout values for Emulex FC HBAs:

Property type	Property value
LinkTimeOut	1
NodeTimeOut	10

   QLogic FC

   The timeout values for QLogic FC HBAs:

Property type	Property value
LinkDownTimeOut	1
PortDownRetryCount	10

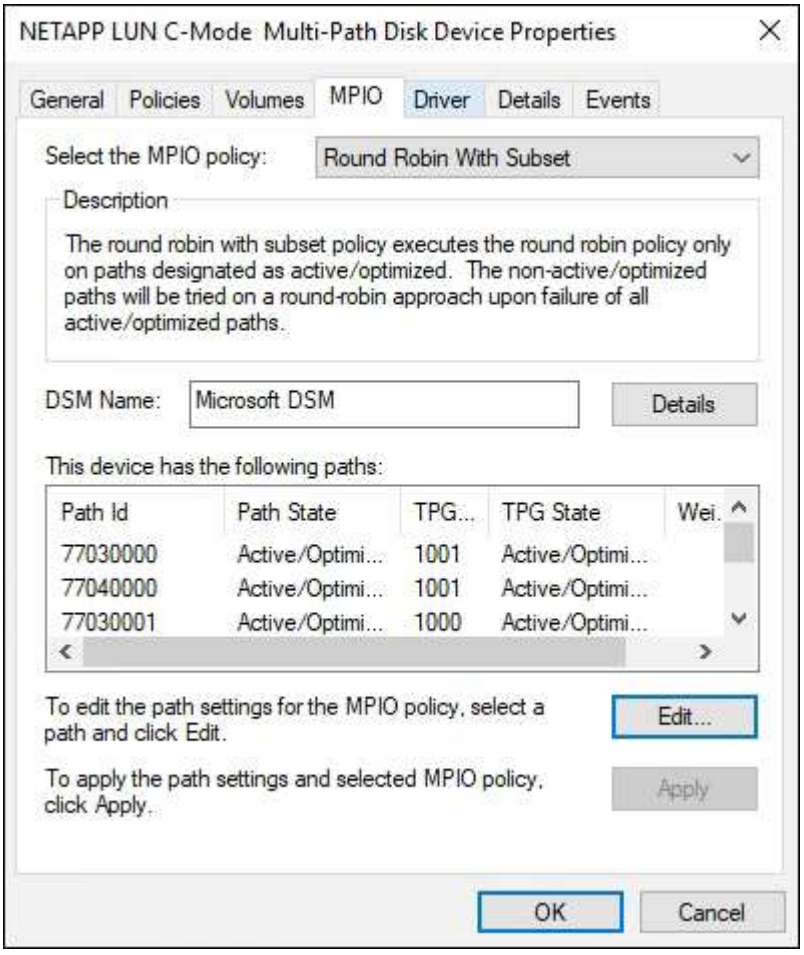
3. Verify the path status for your ONTAP LUNs:

   Depending on your SAN configuration, the host uses ASA, AFF, or FAS configurations to access ONTAP LUNs. These configurations shouldn't require more than four paths to access a single ONTAP LUN. More than four paths can cause problems during storage failure.

   The following example outputs show the correct settings for ONTAP LUNs for an ASA, AFF, or FAS configuration.

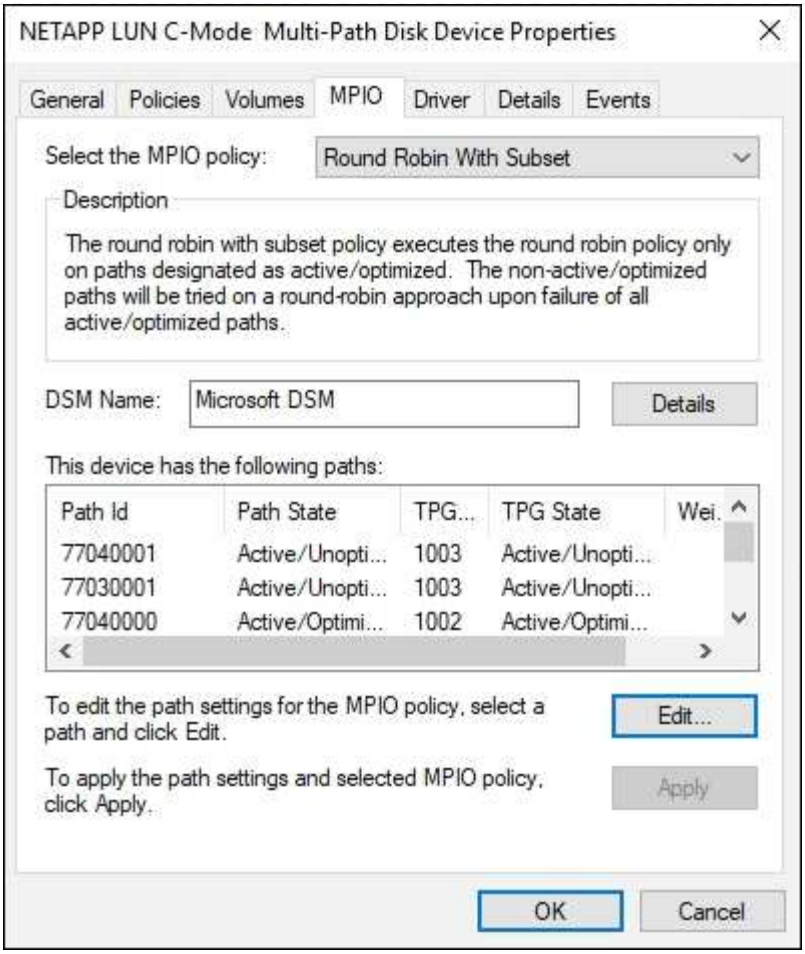
ASA configuration

An ASA configuration should have one group of Active/Optimized paths with single priorities. The controller services the paths and sends I/O on all active paths.



AFF or FAS configuration

An AFF or FAS configuration should have two groups of paths with different priorities. The paths with higher priorities are Active/Optimized and are serviced by the controller where the aggregate is located. The paths with lower priorities are serviced from a different controller. They are active but non-optimized and are only used when optimized paths aren’t available.



Step 5: Review the Known issues

There are no known issues.

What's next?

Learn about the Windows Host Utilities configuration for ONTAP storage

Configure Windows Server 2016 for FCP and iSCSI with ONTAP storage

The Windows Host Utilities enable you to connect Windows hosts to virtual disks (LUNs) on a NetApp SAN. Install the Windows Host Utilities on a Windows Server 2016 host to help you manage FCP and iSCSI protocol operations with ONTAP LUNs.

Step 1: Optionally, enable SAN booting

You can boot the Windows OS using a local boot or a SAN boot. NetApp recommends using a SAN boot to simplify deployment and improve scalability..

SAN boot

If you choose to use SAN booting, it must be supported by your configuration.

Before you begin

Use the Interoperability Matrix Tool to verify that your Windows OS, host bus adapter (HBA), HBA firmware, HBA boot BIOS, and ONTAP version support SAN booting.

Steps

1. Create a SAN boot LUN and map it to the host.
2. Enable SAN booting in the server BIOS for the ports to which the SAN boot LUN is mapped.

   For information on how to enable the HBA BIOS, see your vendor-specific documentation.

3. Verify that the configuration was successful by rebooting the host and verifying that the OS is up and running.

Local boot

Perform a local boot by installing the Windows OS on the local hard disk, for example, on an SSD, SATA, or RAID.

Step 2: Install Windows hotfixes

NetApp recommends installing the **latest cumulative update** available from the Microsoft Update Catalog on the host server.

Steps

1. Download the hotfixes from the Microsoft Update Catalog 2016.

   You need to contact Microsoft support for the hotfixes that aren't available for download from the Microsoft Update Catalog.

2. Follow the instructions provided by Microsoft to install the hotfixes.

   Many hotfixes require a Windows host reboot. You can wait to reboot the host until *after* you install or upgrade the Host Utilities.

Step 3: Install the Windows Host Utilities

The Windows Host Utilities are a set of software programs with documentation that enable you to connect host computers to virtual disks (LUNs) on a NetApp SAN. NetApp recommends downloading and installing the latest Windows Host Utilities to support ONTAP LUN management and help technical support collect configuration data.

For Windows Host Utilities configuration and installation information, see the Windows Host Utilities documentation and select the installation procedure for your Windows Host Utilities version.

Step 4: Confirm the multipath configuration for your host

Install the Microsoft Multipath I/O (MPIO) software and enable multipathing if your Windows host has more than one path to the storage system.

On a Windows system, the two main components in an MPIO solution are the device-specific module (DSM) and the Windows MPIO. MPIO presents one disk to the Windows OS for all paths and the DSM manages path failovers.

If you don't install the MPIO software, the Windows OS might see each path as a separate disk. This can lead to data corruption.

Windows XP or Windows Vista running in a Hyper-V virtual machine doesn't support MPIO.

Steps

1. Install the MPIO software and enable multipathing.
2. When you select MPIO on systems using FC, the Host Utilities installer sets the required timeout values for Emulex and QLogic FC HBAs.

   Emulex FC

   The timeout values for Emulex FC HBAs:

Property type	Property value
LinkTimeOut	1
NodeTimeOut	10

   QLogic FC

   The timeout values for QLogic FC HBAs:

Property type	Property value
LinkDownTimeOut	1
PortDownRetryCount	10

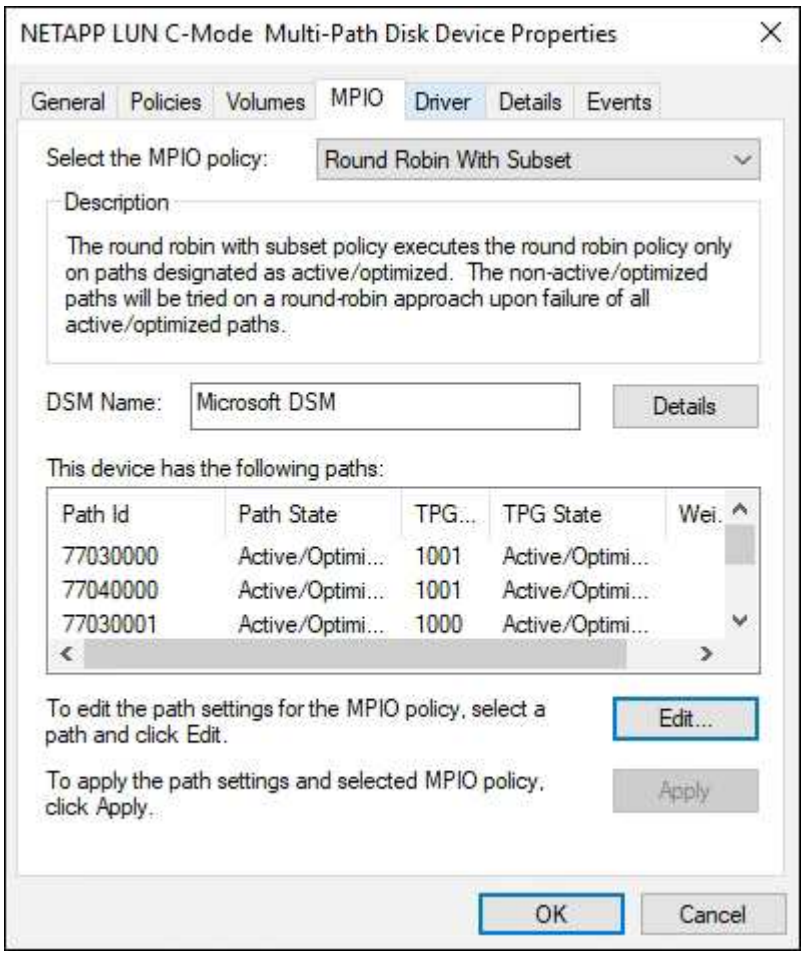
3. Verify the path status for your ONTAP LUNs:

   Depending on your SAN configuration, the host uses ASA, AFF, or FAS configurations to access ONTAP LUNs. These configurations shouldn't require more than four paths to access a single ONTAP LUN. More than four paths can cause problems during storage failure.

   The following example outputs show the correct settings for ONTAP LUNs for an ASA, AFF, or FAS configuration.

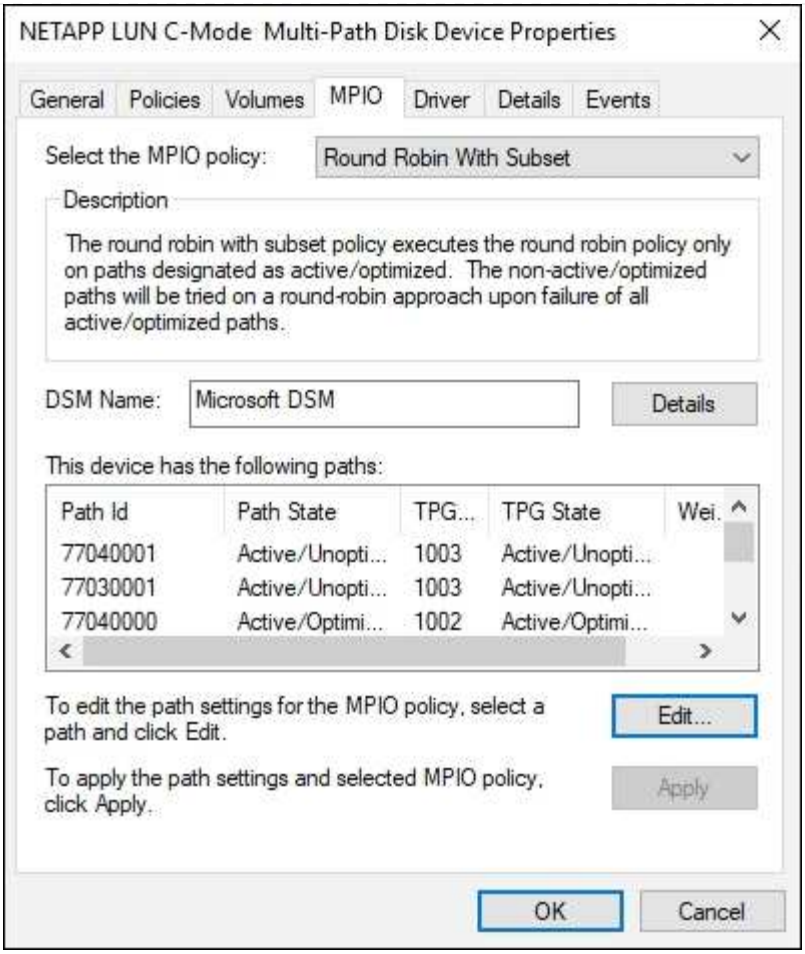
ASA configuration

An ASA configuration should have one group of Active/Optimized paths with single priorities. The controller services the paths and sends I/O on all active paths.



AFF or FAS configuration

An AFF or FAS configuration should have two groups of paths with different priorities. The paths with higher priorities are Active/Optimized and are serviced by the controller where the aggregate is located. The paths with lower priorities are serviced from a different controller. They are active but non-optimized and are only used when optimized paths aren't available.



Step 5: Review the Known issues

There are no known issues.

What's next?

Learn about the Windows Host Utilities configuration for ONTAP storage

Configure Windows Server 2012 R2 for FCP and iSCSI with ONTAP storage

The Windows Host Utilities enable you to connect Windows hosts to virtual disks (LUNs) on a NetApp SAN. Install the Windows Host Utilities on a Windows Server 2012 R2 host to help you manage FCP and iSCSI protocol operations with ONTAP LUNs.

Step 1: Optionally, enable SAN booting

You can boot the Windows OS using a local boot or a SAN boot. NetApp recommends using a SAN boot to simplify deployment and improve scalability..

SAN boot

If you choose to use SAN booting, it must be supported by your configuration.

Before you begin

Use the Interoperability Matrix Tool to verify that your Windows OS, host bus adapter (HBA), HBA firmware, HBA boot BIOS, and ONTAP version support SAN booting.

Steps

1. Create a SAN boot LUN and map it to the host.
2. Enable SAN booting in the server BIOS for the ports to which the SAN boot LUN is mapped.

   For information on how to enable the HBA BIOS, see your vendor-specific documentation.

3. Verify that the configuration was successful by rebooting the host and verifying that the OS is up and running.

Local boot

Perform a local boot by installing the Windows OS on the local hard disk, for example, on an SSD, SATA, or RAID.

Step 2: Install Windows hotfixes

NetApp recommends installing the **latest cumulative update** available from the Microsoft Update Catalog on the host server.

Steps

1. Download the hotfixes from the Microsoft Update Catalog 2012 R2.

   You need to contact Microsoft support for the hotfixes that aren't available for download from the Microsoft Update Catalog.

2. Follow the instructions provided by Microsoft to install the hotfixes.

   Many hotfixes require a Windows host reboot. You can wait to reboot the host until *after* you install or upgrade the Host Utilities.

Step 3: Install the Windows Host Utilities

The Windows Host Utilities are a set of software programs with documentation that enable you to connect host computers to virtual disks (LUNs) on a NetApp SAN. NetApp recommends downloading and installing the latest Windows Host Utilities to support ONTAP LUN management and help technical support collect configuration data.

For Windows Host Utilities configuration and installation information, see the Windows Host Utilities documentation and select the installation procedure for your Windows Host Utilities version.

Step 4: Confirm the multipath configuration for your host

Install the Microsoft Multipath I/O (MPIO) software and enable multipathing if your Windows host has more than one path to the storage system.

On a Windows system, the two main components in an MPIO solution are the device-specific module (DSM) and the Windows MPIO. MPIO presents one disk to the Windows OS for all paths and the DSM manages path failovers.

If you don't install the MPIO software, the Windows OS might see each path as a separate disk. This can lead to data corruption.

Windows XP or Windows Vista running in a Hyper-V virtual machine doesn't support MPIO.

Steps

1. Install the MPIO software and enable multipathing.
2. When you select MPIO on systems using FC, the Host Utilities installer sets the required timeout values for Emulex and QLogic FC HBAs.

   Emulex FC

   The timeout values for Emulex FC HBAs:

Property type	Property value
LinkTimeOut	1
NodeTimeOut	10

   QLogic FC

   The timeout values for QLogic FC HBAs:

Property type	Property value
LinkDownTimeOut	1
PortDownRetryCount	10

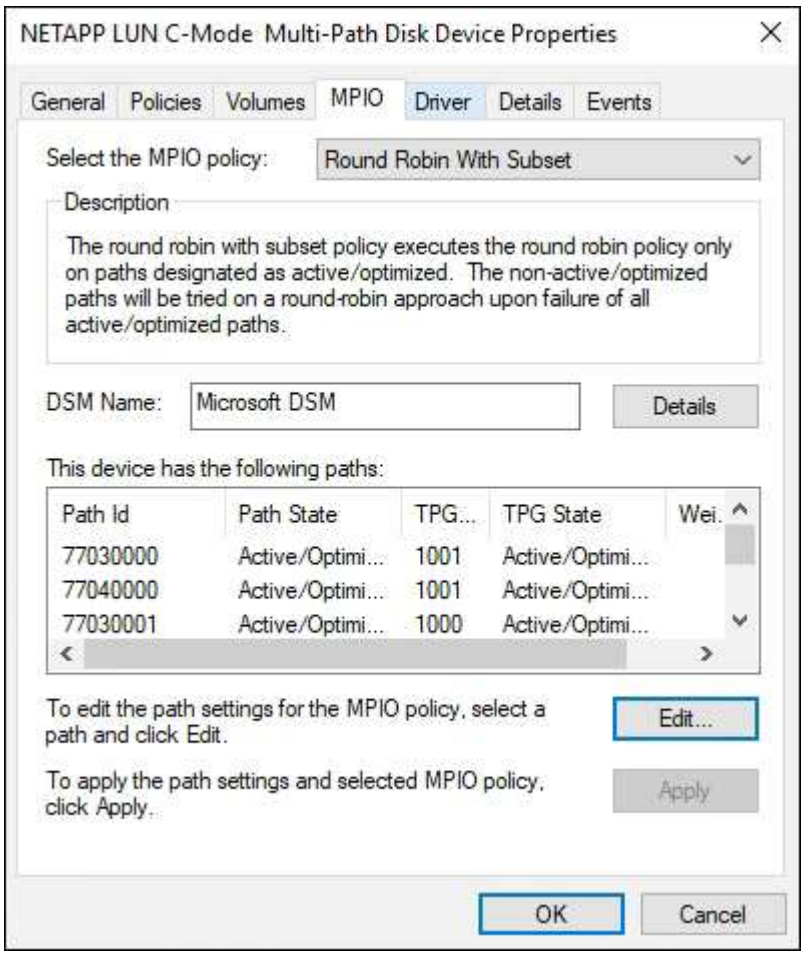
3. Verify the path status for your ONTAP LUNs:

   Depending on your SAN configuration, the host uses ASA, AFF, or FAS configurations to access ONTAP LUNs. These configurations shouldn't require more than four paths to access a single ONTAP LUN. More than four paths can cause problems during storage failure.

   The following example outputs show the correct settings for ONTAP LUNs for an ASA, AFF, or FAS configuration.

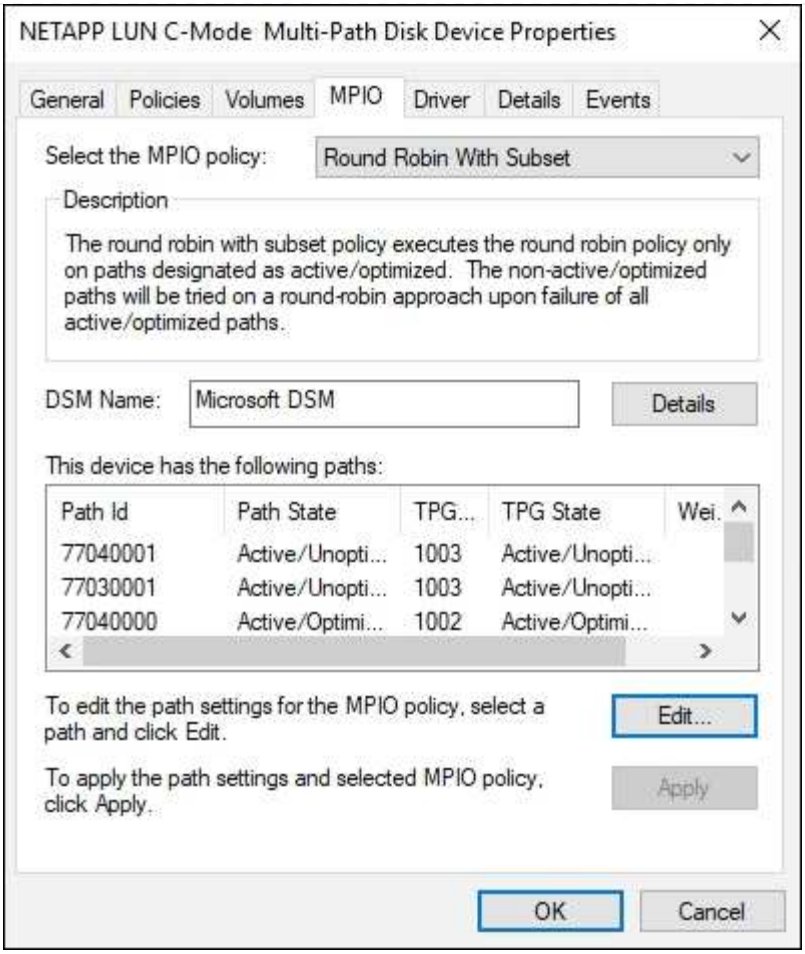
ASA configuration

An ASA configuration should have one group of Active/Optimized paths with single priorities. The controller services the paths and sends I/O on all active paths.



AFF or FAS configuration

An AFF or FAS configuration should have two groups of paths with different priorities. The paths with higher priorities are Active/Optimized and are serviced by the controller where the aggregate is located. The paths with lower priorities are serviced from a different controller. They are active but non-optimized and are only used when optimized paths aren't available.



Step 5: Review the Known issues

There are no known issues.

What's next?

Learn about the Windows Host Utilities configuration for ONTAP storage

Configure hosts with NVMe-oF

Learn about configuring ONTAP SAN hosts with NVMe-oF

You can configure supported SAN hosts to use the NVMe over Fabrics (NVMe-oF) protocol with ONTAP as the storage target. NVMe-oF includes NVMe over Fibre Channel (NVMe/FC) and NVMe over TCP (NVMe/TCP). Depending on your host operating system and ONTAP version, you configure and validate the NVMe/FC or NVMe/TCP protocol, or both on the host.

Related information

- NetApp Knowledge Base: Learn about NVMe-oF.

Configure AIX with NVMe-oF for ONTAP storage

The IBM AIX and Virtual I/O Server (VIOS)/PowerVM hosts supports the NVMe/FC protocol with Asymmetric Namespace Access (ANA). ANA is equivalent to asymmetric logical unit access (ALUA) multipathing in iSCSI and FCP environments.

For additional details on supported configurations, see the Interoperability Matrix Tool (IMT).

About this task

You can use the following support and features with the NVMe-oF host configuration for AIX hosts. You should also review the known limitations before starting the configuration process.

- Support available:
  - Beginning with ONTAP 9.13.1, NVMe/FC support is added for IBM AIX 7.2 TL5 SP6, AIX 7.3 TL1 SP2, and VIOS 3.1.4.21 with SAN boot support for both physical and virtual stacks. See the IBM documentation for more information on setting up SAN boot support.
  - NVMe/FC is supported with Power9 and Power10 IBM servers.
  - A separate PCM (Path Control Module), such as Host Utilities for AIX SCSI Multipath I/O (MPIO) support, isn't required for NVMe devices.
  - Virtualization support with NetApp (VIOS/PowerVM) is introduced with VIOS 3.1.4.21. This is *only* supported through NPIV (N_PortID Virtualization) storage virtualization mode using the Power10 IBM server.
- Known limitations:
  - Qlogic/Marvel 32G FC HBAs on an AIX host doesn't support NVMe/FC.
  - SAN boot isn't supported for NVMe/FC devices using Power9 IBM server.

Before you begin

- Verify that you have 32GB FC Emulex adapters (EN1A, EN1B, EN1L, EN1M) or 64GB FC adapters (EN1N, EN1P) with adapter firmware 12.4.257.30 and later versions.
- If you have a MetroCluster configuration, NetApp recommends changing the AIX NVMe/FC default APD (All Path Down) time for supporting MetroCluster unplanned switchover events to avoid the AIX operating system enforcing a shorter I/O timeout. For additional information and the recommended changes to default settings, refer to NetApp Bugs Online - 1553249.

- Depending on your AIX version, the Asymmetric Namespace Access Transition Timeout (ANATT) for the AIX host OS is 30 seconds or 60 seconds by default. If the ANATT default for your host is 30 seconds, you need to install an IBM Interim Fix (ifix) from the IBM website that sets the ANATT to 60 seconds to ensure that all ONTAP workflows are non-disruptive.

  For NVMe/FC AIX support, you must install an ifix on the GA version of the AIX OS. The ifix isn't required for the VIOS/PowerVM OS.

  You need to install the ifixes on an AIX version with no previously installed ifixes related to `devices.pciex.pciexclass.010802.rte` on the system. Previously installed ifixes can conflict with the new installation.

  Set ANATT to 60 seconds

  The default ANATT for the AIX level 72-TL5-SP6-2320 and AIX level 73-TL1-SP2-2320 releases is 30 seconds. IBM provides an ifix that sets the ANATT to 60 seconds. The ifix is available through IBM case ID TS018079082 and you can install it for the following AIX releases:

  - For AIX level 72-TL5-SP6-2320, install the `IJ46710s6a.230509.epkg.Z` package.
  - For AIX level 73-TL1-SP2-2320, install the `IJ46711s2a.230509.epkg.Z` package.

  Default ANATT is 60 seconds

  The default ANATT is 60 seconds for the following AIX releases:

  - AIX level 73-TL2-SP3-2446
  - AIX level 73-TL2-SP2-2420
  - AIX level 72-TL5-SP8-2420

  Optionally, set ANATT to 120 seconds

  IBM provides an ifix that sets the ANATT to 120 seconds. When you set the ANATT to 120 seconds, it enhances performance during ONTAP storage failover events. The ifix is available through IBM case ID TS012877410 and you can install it for the following AIX releases:

  - For AIX level 73-TL3-SP0-2446, install the `IJ53487s0a.250130.epkg.Z` package.
  - For AIX level 72-TL5-SP9-2446, install the `IJ53445s9a.250130.epkg.Z` package.

  The minimum server firmware version for Power9 servers for NVMe/FC support is FW 950.

  The minimum server firmware version for Power10 servers for NVMe/FC support is FW 1010.

  For more information on managing ifixes, see Managing Interim Fixes on AIX.

Step 1: Confirm the multipath configuration for your host

When you install the AIX OS, IBM MPIO used for NVMe multipathing is enabled by default.

Steps

1. Verify that NVMe multipathing is enabled:

```
lsmpio -l hdisk1
```

Show example

```
name     path_id  status   path_status  parent  connection
hdisk1   8        Enabled  Sel,Opt       nvme12  fcnvme0, 9
hdisk1   9        Enabled  Sel,Non       nvme65  fcnvme1, 9
hdisk1   10       Enabled  Sel,Opt       nvme37  fcnvme1, 9
hdisk1   11       Enabled  Sel,Non       nvme60  fcnvme0, 9
```

Step 2: Configure NVMe/FC

You need to configure NVMe/FC for Broadcom/Emulex adapters on VIOS because the NVMe/FC protocol support is disabled in the Virtual Fibre Channel (vFC) on VIOS. The NVMe/FC protocol support is enabled in the physical FC by default.

Steps

1. Verify that you are using the supported adapter.
2. Retrieve a list of virtual adapters:

```
lsmap -all -npiv
```

Show example

```
Name          Physloc                            ClntID ClntName
ClntOS
------------- ---------------------------------- ------
-------------- -------
vfchost0      U9105.22A.785DB61-V2-C2                4 s1022-iop-
mcc- AIX
Status:LOGGED_IN
FC name:fcs4                    FC loc code:U78DA.ND0.WZS01UY-P0-C7-
T0
Ports logged in:3
Flags:0xea<LOGGED_IN,STRIP_MERGE,SCSI_CLIENT,NVME_CLIENT>
VFC client name:fcs0            VFC client DRC:U9105.22A.785DB61-V4-
C2
```

3. Enable support for the NVMe/FC protocol on an adapter by running the `ioscli vfcctrl` command on the VIOS:

```
vfcctrl -enable -protocol nvme -vadapter vfchost0
```

Example output

```
The "nvme" protocol for "vfchost0" is enabled.
```

4. Verify that the support has been enabled on the adapter:

```
lsattr -El vfchost0
```

Show example

```
alt_site_wwpn       WWPN to use - Only set after migration   False
current_wwpn  0     WWPN to use - Only set after migration   False
enable_nvme   yes   Enable or disable NVME protocol for NPIV True
label               User defined label                       True
limit_intr    false Limit NPIV Interrupt Sources             True
map_port      fcs4  Physical FC Port                         False
num_per_nvme  0     Number of NPIV NVME queues per range     True
num_per_range 0     Number of NPIV SCSI queues per range     True
```

5. Enable the NVMe/Fc protocol for all adapters:
   a. Change the `dflt_enabl_nvme` attribute value of `viosnpiv0` pseudo device to `yes`.
   b. Set the `enable_nvme` attribute value to `yes` for all the VFC host devices.

```
chdev -l viosnpiv0 -a dflt_enabl_nvme=yes
```

```
lsattr -El viosnpiv0
```

Show example

```
bufs_per_cmd    10  NPIV Number of local bufs per cmd
True
dflt_enabl_nvme yes Default NVME Protocol setting for a new NPIV
adapter True
num_local_cmds  5   NPIV Number of local cmds per channel
True
num_per_nvme    8   NPIV Number of NVME queues per range
True
num_per_range   8   NPIV Number of SCSI queues per range
True
secure_va_info  no  NPIV Secure Virtual Adapter Information
True
```

6. Enable the NVMe/Fc protocol for selected adapters by changing the `enable_nvme` value of the VFC host device attribute to `yes`.
7. Verify that `FC-NVMe Protocol Device` has been created on the server:

```
lsdev |grep fcnvme
```

Exmaple output

```
fcnvme0       Available 00-00-02    FC-NVMe Protocol Device
fcnvme1       Available 00-01-02    FC-NVMe Protocol Device
```

8. Record the host NQN from the server:

```
lsattr -El fcnvme0
```

Show example

```
attach     switch
How this adapter is connected  False
autoconfig available
Configuration State            True
host_nqn   nqn.2014-08.org.nvmexpress:uuid:64e039bd-27d2-421c-858d-
8a378dec31e8 Host NQN (NVMe Qualified Name) True
```

```
lsattr -El fcnvme1
```

Show example

```
attach     switch
How this adapter is connected  False
autoconfig available
Configuration State            True
host_nqn   nqn.2014-08.org.nvmexpress:uuid:64e039bd-27d2-421c-858d-
8a378dec31e8 Host NQN (NVMe Qualified Name) True
```

9. Check the host NQN and verify that it matches the host NQN string for the corresponding subsystem on the ONTAP array:

```
vserver nvme subsystem host show -vserver vs_s922-55-lpar2
```

Example output

```
Vserver         Subsystem                Host NQN
------- ---------
------------------------------------------------------------
vs_s922-55-lpar2 subsystem_s922-55-lpar2 nqn.2014-
08.org.nvmexpress:uuid:64e039bd-27d2-421c-858d-8a378dec31e8
```

10. Verify that the initiator ports are up and running and you can see the target LIFs.

Step 3: Validate NVMe/FC

Verify that the ONTAP namespaces are correct for the NVMe/FC configuration.

Steps

1. Verify that the ONTAP namespaces correctly reflect on the host:

```
lsdev -Cc disk |grep NVMe
```

Example output

```
hdisk1  Available 00-00-02 NVMe 4K Disk
```

2. Optionally, check the multipathing status:

```
lsmpio -l hdisk1
```

Show example

```
name     path_id  status   path_status  parent  connection
hdisk1   8        Enabled  Sel,Opt      nvme12  fcnvme0, 9
hdisk1   9        Enabled  Sel,Non      nvme65  fcnvme1, 9
hdisk1   10       Enabled  Sel,Opt      nvme37  fcnvme1, 9
hdisk1   11       Enabled  Sel,Non      nvme60  fcnvme0, 9
```

Step 4: Review the known issues

The NVMe/FC host configuration for AIX with ONTAP storage has the following known issues:

Burt ID	Title	Description
1553249	AIX NVMe/FC default APD time to be modified for supporting MCC Unplanned Switchover events	By default, AIX operating systems use an all path down (APD) timeout value of 20sec for NVMe/FC. However, ONTAP MetroCluster automatic unplanned switchover (AUSO) and TieBreaker initiated switchover workflows might take a little longer than the APD timeout window, causing I/O errors.
1546017	AIX NVMe/FC caps ANATT at 60s, instead of 120s as advertised by ONTAP	ONTAP advertises the ANA(asymmetric namespace access) transition timeout in controller identify at 120sec. Currently, with ifix, AIX reads the ANA transition timeout from controller identify, but effectively clamps it to 60sec if it is over that limit.
1541386	AIX NVMe/FC hits EIO after ANATT expiry	For any storage failover (SFO) events, if the ANA(asymmetric namespace access) transitioning exceeds the ANA transition timeout cap on a given path, the AIX NVMe/FC host fails with an I/O error despite having alternate healthy paths available to the namespace.
1541380	AIX NVMe/FC waits for half/full ANATT to expire before resuming I/O after ANA AEN	IBM AIX NVMe/FC does not support some Asynchronous notifications (AENs) that ONTAP publishes. This sub-optimal ANA handling will result in sub optimal performance during SFO operations.

Step 5: Troubleshoot

Before troubleshooting any NVMe/FC failures, verify that you are running a configuration that is compliant with the IMT specifications. If you continue to have issues, contact NetApp support.

ESXi

NVMe-oF Host Configuration for ESXi 8.x with ONTAP

You can configure NVMe over Fabrics (NVMe-oF) on initiator hosts running ESXi 8.x and ONTAP as the target.

Supportability

- Beginning with ONTAP 9.16.1 space allocation is enabled by default for all newly created NVMe namespaces.
- Beginning with ONTAP 9.9.1 P3, NVMe/FC protocol is supported for ESXi 8 and later.
- Beginning with ONTAP 9.10.1, NVMe/TCP protocol is supported for ONTAP.

Features

- ESXi initiator hosts can run both NVMe/FC and FCP traffic through the same adapter ports. See the Hardware Universe for a list of supported FC adapters and controllers. See the NetApp Interoperability Matrix Tool for the most current list of supported configurations and versions.
- For ESXi 8.0 and later releases, HPP (high performance plugin) is the default plugin for NVMe devices.

Known limitations

- RDM mapping is not supported.

Enable NVMe/FC

NVMe/FC is enabled by default in vSphere releases.

Verify host NQN

You must check the ESXi host NQN string and verify that it matches with the host NQN string for the corresponding subsystem on the ONTAP array.

```
# esxcli nvme info get
```

Example output:

```
Host NQN: nqn.2014-08.org.nvmexpress:uuid:62a19711-ba8c-475d-c954-
0000c9f1a436
```

```
# vserver nvme subsystem host show -vserver nvme_fc
```

Example output:

```
Vserver Subsystem Host NQN
------- ---------
-------------------------------------------------------------
nvme_fc nvme_ss  nqn.2014-08.org.nvmexpress:uuid:62a19711-ba8c-475d-c954-
0000c9f1a436
```

If the host NQN strings do not match, you should use the `vserver nvme subsystem host add` command to update the correct host NQN string on your corresponding ONTAP NVMe subsystem.

Configure Broadcom/Emulex and Marvell/Qlogic

The `lpfc` driver and the `qlnativefc` driver in vSphere 8.x have the NVMe/FC capability enabled by default.

See Interoperability Matrix Tool to check whether the configuration is supported with the driver or firmware.

Validate NVMe/FC

You can use the following procedure to validate NVMe/FC.

Steps

1. Verify that the NVMe/FC adapter is listed on the ESXi host:

   ```
   # esxcli nvme adapter list
   ```

   Example output:

   ```
   Adapter  Adapter Qualified Name           Transport Type  Driver
   Associated Devices
   -------  -------------------------------  --------------  ----------
   ------------------
   vmhba64  aqn:lpfc:100000109b579f11        FC              lpfc
   vmhba65  aqn:lpfc:100000109b579f12        FC              lpfc
   vmhba66  aqn:qlnativefc:2100f4e9d456e286  FC              qlnativefc
   vmhba67  aqn:qlnativefc:2100f4e9d456e287  FC              qlnativefc
   ```

2. Verify that the NVMe/FC namespaces are correctly created:

   The UUIDs in the following example represent the NVMe/FC namespace devices.

```
# esxcfg-mpath -b
uuid.116cb7ed9e574a0faf35ac2ec115969d : NVMe Fibre Channel Disk
(uuid.116cb7ed9e574a0faf35ac2ec115969d)
   vmhba64:C0:T0:L5 LUN:5 state:active fc Adapter: WWNN:
20:00:00:24:ff:7f:4a:50 WWPN: 21:00:00:24:ff:7f:4a:50  Target: WWNN:
20:04:d0:39:ea:3a:b2:1f WWPN: 20:05:d0:39:ea:3a:b2:1f
   vmhba64:C0:T1:L5 LUN:5 state:active fc Adapter: WWNN:
20:00:00:24:ff:7f:4a:50 WWPN: 21:00:00:24:ff:7f:4a:50  Target: WWNN:
20:04:d0:39:ea:3a:b2:1f WWPN: 20:07:d0:39:ea:3a:b2:1f
   vmhba65:C0:T1:L5 LUN:5 state:active fc Adapter: WWNN:
20:00:00:24:ff:7f:4a:51 WWPN: 21:00:00:24:ff:7f:4a:51  Target: WWNN:
20:04:d0:39:ea:3a:b2:1f WWPN: 20:08:d0:39:ea:3a:b2:1f
   vmhba65:C0:T0:L5 LUN:5 state:active fc Adapter: WWNN:
20:00:00:24:ff:7f:4a:51 WWPN: 21:00:00:24:ff:7f:4a:51  Target: WWNN:
20:04:d0:39:ea:3a:b2:1f WWPN: 20:06:d0:39:ea:3a:b2:1f
```

In ONTAP 9.7, the default block size for an NVMe/FC namespace is 4K. This default size is not compatible with ESXi. Therefore, when creating namespaces for ESXi, you must set the namespace block size as **512B**. You can do this using the `vserver nvme namespace create` command.

Example,

```
vserver nvme namespace create -vserver vs_1 -path
/vol/nsvol/namespace1 -size 100g -ostype vmware -block-size 512B
```

Refer to the ONTAP 9 Command man pages for additional details.

3. Verify the status of the individual ANA paths of the respective NVMe/FC namespace devices:

```
# esxcli storage hpp path list -d uuid.df960bebb5a74a3eaaa1ae55e6b3411d

fc.20000024ff7f4a50:21000024ff7f4a50-
fc.2004d039ea3ab21f:2005d039ea3ab21f-
uuid.df960bebb5a74a3eaaa1ae55e6b3411d
   Runtime Name: vmhba64:C0:T0:L3
   Device: uuid.df960bebb5a74a3eaaa1ae55e6b3411d
   Device Display Name: NVMe Fibre Channel Disk
(uuid.df960bebb5a74a3eaaa1ae55e6b3411d)
   Path State: active unoptimized
   Path Config: {ANA_GRP_id=4,ANA_GRP_state=ANO,health=UP}

fc.20000024ff7f4a51:21000024ff7f4a51-
fc.2004d039ea3ab21f:2008d039ea3ab21f-
uuid.df960bebb5a74a3eaaa1ae55e6b3411d
   Runtime Name: vmhba65:C0:T1:L3
   Device: uuid.df960bebb5a74a3eaaa1ae55e6b3411d
   Device Display Name: NVMe Fibre Channel Disk
(uuid.df960bebb5a74a3eaaa1ae55e6b3411d)
   Path State: active
   Path Config: {ANA_GRP_id=4,ANA_GRP_state=AO,health=UP}

fc.20000024ff7f4a51:21000024ff7f4a51-
fc.2004d039ea3ab21f:2006d039ea3ab21f-
uuid.df960bebb5a74a3eaaa1ae55e6b3411d
   Runtime Name: vmhba65:C0:T0:L3
   Device: uuid.df960bebb5a74a3eaaa1ae55e6b3411d
   Device Display Name: NVMe Fibre Channel Disk
(uuid.df960bebb5a74a3eaaa1ae55e6b3411d)
   Path State: active unoptimized
   Path Config: {ANA_GRP_id=4,ANA_GRP_state=ANO,health=UP}

fc.20000024ff7f4a50:21000024ff7f4a50-
fc.2004d039ea3ab21f:2007d039ea3ab21f-
uuid.df960bebb5a74a3eaaa1ae55e6b3411d
   Runtime Name: vmhba64:C0:T1:L3
   Device: uuid.df960bebb5a74a3eaaa1ae55e6b3411d
   Device Display Name: NVMe Fibre Channel Disk
(uuid.df960bebb5a74a3eaaa1ae55e6b3411d)
   Path State: active
   Path Config: {ANA_GRP_id=4,ANA_GRP_state=AO,health=UP}
```

Configure NVMe/TCP

In ESXi 8.x, the required NVMe/TCP modules are loaded by default. To configure the network and the

NVMe/TCP adapter, refer to the VMware vSphere documentation.

Validate NVMe/TCP

You can use the following procedure to validate NVMe/TCP.

Steps

1. Verify the status of the NVMe/TCP adapter:

```
esxcli nvme adapter list
```

Example output:

```
Adapter  Adapter Qualified Name           Transport Type  Driver
Associated Devices
-------  -------------------------------  --------------  -------
------------------
vmhba65  aqn:nvmetcp:ec-2a-72-0f-e2-30-T  TCP             nvmetcp
vmnic0
vmhba66  aqn:nvmetcp:34-80-0d-30-d1-a0-T  TCP             nvmetcp
vmnic2
vmhba67  aqn:nvmetcp:34-80-0d-30-d1-a1-T  TCP             nvmetcp
vmnic3
```

2. Retrieve a list of NVMe/TCP connections:

```
esxcli nvme controller list
```

Example output:

```
Name                                                              Controller Number
Adapter  Transport Type  Is Online  Is VVOL
------------------------------------------------------------------------------
--------------------------------  -----------------  -------
nqn.2014-08.org.nvmexpress.discovery#vmhba64#192.168.100.166:8009  256
vmhba64  TCP                  true    false
nqn.1992-
08.com.netapp:sn.89bb1a28a89a11ed8a88d039ea263f93:subsystem.nvme_ss#vmhb
a64#192.168.100.165:4420 258  vmhba64  TCP  true    false
nqn.1992-
08.com.netapp:sn.89bb1a28a89a11ed8a88d039ea263f93:subsystem.nvme_ss#vmhb
a64#192.168.100.168:4420 259  vmhba64  TCP  true    false
nqn.1992-
08.com.netapp:sn.89bb1a28a89a11ed8a88d039ea263f93:subsystem.nvme_ss#vmhb
a64#192.168.100.166:4420 260  vmhba64  TCP  true    false
nqn.2014-08.org.nvmexpress.discovery#vmhba64#192.168.100.165:8009  261
vmhba64  TCP                  true    false
nqn.2014-08.org.nvmexpress.discovery#vmhba65#192.168.100.155:8009  262
vmhba65  TCP                  true    false
nqn.1992-
08.com.netapp:sn.89bb1a28a89a11ed8a88d039ea263f93:subsystem.nvme_ss#vmhb
a64#192.168.100.167:4420 264  vmhba64  TCP  true    false
```

3. Retrieve a list of the number of paths to an NVMe namespace:

```
esxcli storage hpp path list -d uuid.f4f14337c3ad4a639edf0e21de8b88bf
```

Example output:

```
tcp.vmnic2:34:80:0d:30:ca:e0-tcp.192.168.100.165:4420-
uuid.f4f14337c3ad4a639edf0e21de8b88bf
   Runtime Name: vmhba64:C0:T0:L5
   Device: uuid.f4f14337c3ad4a639edf0e21de8b88bf
   Device Display Name: NVMe TCP Disk
(uuid.f4f14337c3ad4a639edf0e21de8b88bf)
   Path State: active
   Path Config: {ANA_GRP_id=6,ANA_GRP_state=AO,health=UP}

tcp.vmnic2:34:80:0d:30:ca:e0-tcp.192.168.100.168:4420-
uuid.f4f14337c3ad4a639edf0e21de8b88bf
   Runtime Name: vmhba64:C0:T3:L5
   Device: uuid.f4f14337c3ad4a639edf0e21de8b88bf
   Device Display Name: NVMe TCP Disk
(uuid.f4f14337c3ad4a639edf0e21de8b88bf)
   Path State: active unoptimized
   Path Config: {ANA_GRP_id=6,ANA_GRP_state=ANO,health=UP}

tcp.vmnic2:34:80:0d:30:ca:e0-tcp.192.168.100.166:4420-
uuid.f4f14337c3ad4a639edf0e21de8b88bf
   Runtime Name: vmhba64:C0:T2:L5
   Device: uuid.f4f14337c3ad4a639edf0e21de8b88bf
   Device Display Name: NVMe TCP Disk
(uuid.f4f14337c3ad4a639edf0e21de8b88bf)
   Path State: active unoptimized
   Path Config: {ANA_GRP_id=6,ANA_GRP_state=ANO,health=UP}

tcp.vmnic2:34:80:0d:30:ca:e0-tcp.192.168.100.167:4420-
uuid.f4f14337c3ad4a639edf0e21de8b88bf
   Runtime Name: vmhba64:C0:T1:L5
   Device: uuid.f4f14337c3ad4a639edf0e21de8b88bf
   Device Display Name: NVMe TCP Disk
(uuid.f4f14337c3ad4a639edf0e21de8b88bf)
   Path State: active
   Path Config: {ANA_GRP_id=6,ANA_GRP_state=AO,health=UP}
```

NVMe deallocate

The NVMe deallocate command is supported for ESXi 8.0u2 and later with ONTAP 9.16.1 and later.

Deallocate support is always enabled for NVMe namespaces. Deallocate also allows the guest OS to perform 'UNMAP' (sometimes called 'TRIM') operations on VMFS datastores. Deallocate operations allow a host to identify blocks of data that are no longer required because they no longer contain valid data. The storage system can then remove those data blocks so that the space can be consumed elsewhere.

Steps

1. On your ESXi host, verify the setting for DSM deallocate with TP4040 support:

   ```
   esxcfg-advcfg -g /Scsi/NVmeUseDsmTp4040
   ```

   The expected value is 0.

2. Enable the setting for DSM deallocate with TP4040 support:

   ```
   esxcfg-advcfg -s 1 /Scsi/NvmeUseDsmTp4040
   ```

3. Verify that the setting for DSM deallocate with TP4040 support is enabled:

   ```
   esxcfg-advcfg -g /Scsi/NVmeUseDsmTp4040
   ```

   The expected value is 1.

For more information on NVMe deallocate in VMware vSphere, refer to Storage Space Reclamation in vSphere

Known issues

The NVMe-oF host configuration for ESXi 8.x with ONTAP has the following known issues:

NetApp Bug ID	Title	Description
1420654	ONTAP node non-operational when NVMe/FC protocol is used with ONTAP version 9.9.1	ONTAP 9.9.1 has introduced support for the NVMe "abort" command. When ONTAP receives the "abort" command to abort an NVMe fused command that is waiting for its partner command, an ONTAP node disruption occurs. The issue is noticed only with hosts that use NVMe fused commands (for example, ESX) and Fibre Channel (FC) transport.
1543660	I/O error occurs when Linux VMs using vNVMe adapters encounter a long all paths down (APD) window	Linux VMs running vSphere 8.x and later and using virtual NVMe (vNVME) adapters encounter an I/O error because the vNVMe retry operation is disabled by default. To avoid a disruption on Linux VMs running older kernels during an all paths down (APD) or a heavy I/O load, VMware has introduced a tunable "VSCSIDisableNvmeRetry" to disable the vNVMe retry operation.

Related information

VMware vSphere with ONTAP
VMware vSphere 5.x, 6.x and 7.x support with NetApp MetroCluster (2031038)
VMware vSphere 6.x and 7.x support with NetApp SnapMirror active sync

NVMe-oF Host Configuration for ESXi 7.x with ONTAP

You can configure NVMe over Fabrics (NVMe-oF) on initiator hosts running ESXi 7.x and ONTAP as the target.

Supportability

- Beginning with ONTAP 9.7, NVMe over Fibre Channel (NVMe/FC) support is added for VMWare vSphere releases.
- Beginning with 7.0U3c, NVMe/TCP feature is supported for ESXi Hypervisor.
- Beginning with ONTAP 9.10.1, NVMe/TCP feature is supported for ONTAP.

Features

- ESXi initiator host can run both NVMe/FC and FCP traffic through the same adapter ports. See the Hardware Universe for a list of supported FC adapters and controllers. See the Interoperability Matrix Tool for the current list of supported configurations and versions.
- Beginning with ONTAP 9.9.1 P3, NVMe/FC feature is supported for ESXi 7.0 update 3.
- For ESXi 7.0 and later releases, HPP (high performance plugin) is the default plugin for NVMe devices.

Known limitations

The following configurations are not supported:

- RDM mapping
- VVols

Enable NVMe/FC

1. Check the ESXi host NQN string and verify that it matches with the host NQN string for the corresponding subsystem on the ONTAP array:

```
# esxcli nvme  info get
Host NQN: nqn.2014-08.com.vmware:nvme:nvme-esx

# vserver nvme subsystem host show -vserver vserver_nvme
  Vserver Subsystem             Host NQN
  ------- ------------------ ----------------------------------------
  vserver_nvme ss_vserver_nvme nqn.2014-08.com.vmware:nvme:nvme-esx
```

Configure Broadcom/Emulex

1. Check whether the configuration is supported with required driver/firmware by referring to Interoperability Matrix Tool.
2. Set the lpfc driver parameter `lpfc_enable_fc4_type=3` for enabling NVMe/FC support in the `lpfc` driver and reboot the host.

Starting with vSphere 7.0 update 3, the `brcmnvmefc` driver is no longer available. Therefore, the `lpfc` driver now includes the NVMe over Fibre Channel (NVMe/FC) functionality previously delivered with the `brcmnvmefc` driver.

The `lpfc_enable_fc4_type=3` parameter is set by default for the LPe35000-series adapters. You must perform the following command to set it manually for LPe32000-series and LPe31000-series adapters.

```
# esxcli system module parameters set -m lpfc -p lpfc_enable_fc4_type=3

#esxcli system module parameters list  -m lpfc | grep lpfc_enable_fc4_type
lpfc_enable_fc4_type              int     3      Defines what FC4 types
are supported

#esxcli storage core adapter list
HBA Name  Driver   Link State  UID
Capabilities         Description
--------  -------  ----------  ------------------------------------
------------------  -----------
vmhba1    lpfc     link-up     fc.200000109b95456f:100000109b95456f
Second Level Lun ID  (0000:86:00.0) Emulex Corporation Emulex LPe36000
Fibre Channel Adapter    FC HBA
vmhba2    lpfc     link-up     fc.200000109b954570:100000109b954570
Second Level Lun ID  (0000:86:00.1) Emulex Corporation Emulex LPe36000
Fibre Channel Adapter    FC HBA
vmhba64   lpfc     link-up     fc.200000109b95456f:100000109b95456f
(0000:86:00.0) Emulex Corporation Emulex LPe36000 Fibre Channel Adapter
NVMe HBA
vmhba65   lpfc     link-up     fc.200000109b954570:100000109b954570
(0000:86:00.1) Emulex Corporation Emulex LPe36000 Fibre Channel Adapter
NVMe HBA
```

Configure Marvell/QLogic

Steps

1. Check whether configuration is supported with required driver/firmware by referring to Interoperability Matrix Tool.
2. Set the `qlnativefc` driver parameter `ql2xnvmesupport=1` for enabling NVMe/FC support in the `qlnativefc` driver and reboot the host.

   ```
   # esxcfg-module -s 'ql2xnvmesupport=1' qlnativefc
   ```

   The `qlnativefc` driver parameter is set by default for the Qle 277x-series adapters. You must perform the following command to set it manually for Qle 277x series adapters.

```
esxcfg-module -l | grep qlnativefc
qlnativefc               4    1912
```

3. Check whether nvme is enabled on the adapter:

```
  #esxcli storage core adapter list
HBA Name  Driver      Link State  UID
Capabilities         Description
--------  ----------  ----------  ------------------------------------
------------------  -----------
 vmhba3    qlnativefc  link-up     fc.20000024ff1817ae:21000024ff1817ae
Second Level Lun ID  (0000:5e:00.0) QLogic Corp QLE2742 Dual Port 32Gb
Fibre Channel to PCIe Adapter    FC Adapter
vmhba4    qlnativefc  link-up     fc.20000024ff1817af:21000024ff1817af
Second Level Lun ID  (0000:5e:00.1) QLogic Corp QLE2742 Dual Port 32Gb
Fibre Channel to PCIe Adapter FC Adapter
vmhba64   qlnativefc  link-up     fc.20000024ff1817ae:21000024ff1817ae
(0000:5e:00.0) QLogic Corp QLE2742 Dual Port 32Gb Fibre Channel to PCIe
Adapter  NVMe FC Adapter
vmhba65   qlnativefc  link-up     fc.20000024ff1817af:21000024ff1817af
(0000:5e:00.1) QLogic Corp QLE2742 Dual Port 32Gb Fibre Channel to PCIe
Adapter  NVMe FC Adapter
```

Validate NVMe/FC

1. Verify that NVMe/FC adapter is listed on the ESXi host:

```
# esxcli nvme adapter list

Adapter  Adapter Qualified Name           Transport Type  Driver
Associated Devices
-------  -------------------------------  --------------  ----------
------------------
vmhba64  aqn:qlnativefc:21000024ff1817ae  FC              qlnativefc
vmhba65  aqn:qlnativefc:21000024ff1817af  FC              qlnativefc
vmhba66  aqn:lpfc:100000109b579d9c        FC              lpfc
vmhba67  aqn:lpfc:100000109b579d9d        FC              lpfc
```

2. Verify that the NVMe/FC namespaces are properly created:

   The UUIDs in the following example represent the NVMe/FC namespace devices.

```
# esxcfg-mpath -b
uuid.5084e29a6bb24fbca5ba076eda8ecd7e : NVMe Fibre Channel Disk
(uuid.5084e29a6bb24fbca5ba076eda8ecd7e)
   vmhba65:C0:T0:L1 LUN:1 state:active fc Adapter: WWNN:
20:00:34:80:0d:6d:72:69 WWPN: 21:00:34:80:0d:6d:72:69  Target: WWNN:
20:17:00:a0:98:df:e3:d1 WWPN: 20:2f:00:a0:98:df:e3:d1
   vmhba65:C0:T1:L1 LUN:1 state:active fc Adapter: WWNN:
20:00:34:80:0d:6d:72:69 WWPN: 21:00:34:80:0d:6d:72:69  Target: WWNN:
20:17:00:a0:98:df:e3:d1 WWPN: 20:1a:00:a0:98:df:e3:d1
   vmhba64:C0:T0:L1 LUN:1 state:active fc Adapter: WWNN:
20:00:34:80:0d:6d:72:68 WWPN: 21:00:34:80:0d:6d:72:68  Target: WWNN:
20:17:00:a0:98:df:e3:d1 WWPN: 20:18:00:a0:98:df:e3:d1
   vmhba64:C0:T1:L1 LUN:1 state:active fc Adapter: WWNN:
20:00:34:80:0d:6d:72:68 WWPN: 21:00:34:80:0d:6d:72:68  Target: WWNN:
20:17:00:a0:98:df:e3:d1 WWPN: 20:19:00:a0:98:df:e3:d1
```

In ONTAP 9.7, the default block size for a NVMe/FC namespace is 4K. This default size is not compatible with ESXi. Therefore, when creating namespaces for ESXi, you must set the namespace block size as 512b. You can do this using the `vserver nvme namespace create` command.

Example

```
vserver nvme namespace create -vserver vs_1 -path /vol/nsvol/namespace1 -size
100g -ostype vmware -block-size 512B
```

Refer to the ONTAP 9 Command man pages for additional details.

3. Verify the status of the individual ANA paths of the respective NVMe/FC namespace devices:

```
esxcli storage hpp path list -d uuid.5084e29a6bb24fbca5ba076eda8ecd7e
fc.200034800d6d7268:210034800d6d7268-
fc.201700a098dfe3d1:201800a098dfe3d1-
uuid.5084e29a6bb24fbca5ba076eda8ecd7e
   Runtime Name: vmhba64:C0:T0:L1
   Device: uuid.5084e29a6bb24fbca5ba076eda8ecd7e
   Device Display Name: NVMe Fibre Channel Disk
(uuid.5084e29a6bb24fbca5ba076eda8ecd7e)
   Path State: active
   Path Config: {TPG_id=0,TPG_state=AO,RTP_id=0,health=UP}

fc.200034800d6d7269:210034800d6d7269-
fc.201700a098dfe3d1:201a00a098dfe3d1-
uuid.5084e29a6bb24fbca5ba076eda8ecd7e
   Runtime Name: vmhba65:C0:T1:L1
   Device: uuid.5084e29a6bb24fbca5ba076eda8ecd7e
   Device Display Name: NVMe Fibre Channel Disk
(uuid.5084e29a6bb24fbca5ba076eda8ecd7e)
   Path State: active
   Path Config: {TPG_id=0,TPG_state=AO,RTP_id=0,health=UP}

fc.200034800d6d7269:210034800d6d7269-
fc.201700a098dfe3d1:202f00a098dfe3d1-
uuid.5084e29a6bb24fbca5ba076eda8ecd7e
   Runtime Name: vmhba65:C0:T0:L1
   Device: uuid.5084e29a6bb24fbca5ba076eda8ecd7e
   Device Display Name: NVMe Fibre Channel Disk
(uuid.5084e29a6bb24fbca5ba076eda8ecd7e)
   Path State: active unoptimized
   Path Config: {TPG_id=0,TPG_state=ANO,RTP_id=0,health=UP}

fc.200034800d6d7268:210034800d6d7268-
fc.201700a098dfe3d1:201900a098dfe3d1-
uuid.5084e29a6bb24fbca5ba076eda8ecd7e
   Runtime Name: vmhba64:C0:T1:L1
   Device: uuid.5084e29a6bb24fbca5ba076eda8ecd7e
   Device Display Name: NVMe Fibre Channel Disk
(uuid.5084e29a6bb24fbca5ba076eda8ecd7e)
   Path State: active unoptimized
   Path Config: {TPG_id=0,TPG_state=ANO,RTP_id=0,health=UP}
```

Configure NVMe/TCP

Starting 7.0U3c, the required NVMe/TCP modules will be loaded by default. For configuring the network and the NVMe/TCP adapter, refer to the VMware vSphere documentation.

Validate NVMe/TCP

Steps

1. Verify the status of the NVMe/TCP adapter.

```
[root@R650-8-45:~] esxcli nvme adapter list
Adapter    Adapter Qualified Name
--------- ------------------------------
vmhba64    aqn:nvmetcp:34-80-0d-30-ca-e0-T
vmhba65    aqn:nvmetc:34-80-13d-30-ca-e1-T
list
Transport Type   Driver   Associated Devices
---------------  -------  ------------------
TCP              nvmetcp    vmnzc2
TCP              nvmetcp    vmnzc3
```

2. To list the NVMe/TCP connections, use the following command:

```
[root@R650-8-45:~] esxcli nvme controller list
Name
-----------
nqn.1992-
08.com.netapp:sn.5e347cf68e0511ec9ec2d039ea13e6ed:subsystem.vs_name_tcp_
ss#vmhba64#192.168.100.11:4420
nqn.1992-
08.com.netapp:sn.5e347cf68e0511ec9ec2d039ea13e6ed:subsystem.vs_name_tcp_
ss#vmhba64#192.168.101.11:4420
Controller Number  Adapter   Transport Type   IS Online
-----------------  ---------  ---------------  ---------
1580               vmhba64    TCP              true
1588               vmhba65    TCP              true
```

3. To list the number of paths to an NVMe namespace, use the following command:

```
[root@R650-8-45:~] esxcli storage hpp path list -d
uuid.400bf333abf74ab8b96dc18ffadc3f99
tcp.vmnic2:34:80:0d:30:ca:eo-tcp.unknown-
uuid.400bf333abf74ab8b96dc18ffadc3f99
   Runtime Name: vmhba64:C0:T0:L3
   Device: uuid.400bf333abf74ab8b96dc18ffadc3f99
   Device Display Name: NVMe TCP Disk
(uuid.400bf333abf74ab8b96dc18ffadc3f99)
   Path State: active unoptimized
   Path config: {TPG_id=0,TPG_state=ANO,RTP_id=0,health=UP}

tcp.vmnic3:34:80:0d:30:ca:el-tcp.unknown-
uuid.400bf333abf74ab8b96dc18ffadc3f99
   Runtime Name: vmhba65:C0:T1:L3
   Device: uuid.400bf333abf74ab8b96dc18ffadc3f99
   Device Display Name: NVMe TCP Disk
(uuid.400bf333abf74ab8b96dc18ffadc3f99)
   Path State: active
   Path config: {TPG_id=0,TPG_state=AO,RTP_id=0,health=UP}
```

Known issues

The NVMe-oF host configuration for ESXi 7.x with ONTAP has the following known issues:

NetApp Bug ID	Title	Workaround
1420654	ONTAP node non-operational when NVMe/FC protocol is used with ONTAP version 9.9.1	Check and rectify any network issues in the host fabric. If this does not help, upgrade to a patch that fixes this issue.

Related information

VMware vSphere with ONTAP
VMware vSphere 5.x, 6.x and 7.x support with NetApp MetroCluster (2031038)
VMware vSphere 6.x and 7.x support with NetApp® SnapMirror active sync

Oracle Linux

Learn about ONTAP support and features for Oracle Linux hosts

Features supported for host configuration with NVMe over Fabrics (NVMe-oF) varies based on your version of ONTAP and Oracle Linux.

Feature	Oracle Linux host version	ONTAP version
Secure in-band authentication is supported over NVMe/TCP between a Oracle Linux host and an ONTAP controller	9.4 or later	9.12.1 or later
NVMe/TCP is a fully supported enterprise feature	9.0 or later	9.10.1 or later
NVMe/TCP provides namespaces using the native `nvme-cli` package	8.2 or later	9.10.1 or later
NVMe and SCSI traffic is supported on the same host using NVMe multipath for NVMe-oF namespaces and dm-multipath for SCSI LUNs	7.7 or later	9.4 or later

The NetApp `sanlun` host utility isn't supported for NVMe-oF. Instead, you can use the NetApp plug-in included in the native `nvme-cli` for all NVMe-oF transports.

ONTAP supports the following SAN host features regardless of the ONTAP version running on your system setup.

Feature	Oracle Linux host version
The native udev rule in the `nvme-cli` package provides queue-depth load balancing for NVMe multipathing	9.6 or later
SAN booting is enabled using the NVMe/FC protocol	9.5 or later
In-kernel NVMe multipathing for NVMe namespaces is enabled by default	8.3 or later
The `nvme-cli` package includes auto-connect scripts removing the need for third-party scripts	8.3 or later
The native udev rule in the `nvme-cli` package provides round-robin load balancing for NVMe multipathing	8.3 or later

For details on supported configurations, see the Interoperability Matrix Tool.

What's next

If your Oracle Linux version is ..	Learn about ..
9 series	Configuring NVMe for Oracle Linux 9.x
8 series	Configuring NVMe for Oracle Linux 8.x
7 series	Configuring NVMe for Oracle Linux 7.x

Related information

Learn about managing NVMe protocols

Configure Oracle Linux 9.x with NVMe-oF for ONTAP storage

Oracle Linux hosts support the NVMe over Fibre Channel (NVMe/FC) and NVMe over TCP (NVMe/TCP) protocols with Asymmetric Namespace Access (ANA). ANA provides multipathing functionality equivalent to asymmetric logical unit access (ALUA) in iSCSI and FCP environments.

Learn how to configure NVMe over Fabrics (NVMe-oF) hosts for Oracle Linux 9.x. For more support and feature information, see Oracle Linux ONTAP support and features.

NVMe-oF with Oracle Linux 9.x has the following known limitation:

- The `nvme disconnect-all` command disconnects both root and data filesystems and might lead to system instability. Do not issue this on systems booting from SAN over NVMe-TCP or NVMe-FC namespaces.

Step 1: Optionally, enable SAN booting

You can configure your host to use SAN booting to simplify deployment and improve scalability. Use the Interoperability Matrix Tool to verify that your Linux OS, host bus adapter (HBA), HBA firmware, HBA boot BIOS, and ONTAP version support SAN booting.

Steps

1. Create a NVMe namespace and map it to the host.
2. Enable SAN booting in the server BIOS for the ports to which the SAN boot namespace is mapped.

   For information on how to enable the HBA BIOS, see your vendor-specific documentation.

3. Reboot the host and verify that the OS is up and running.

Step 2: Install Oracle Linux and NVMe software and verify your configuration

Use the following procedure to validate the minimum supported Oracle Linux 9.x software versions.

Steps

1. Install Oracle Linux 9.x on the server. After the installation is complete, verify that you are running the specified Oracle Linux 9.x kernel.

   ```
   uname -r
   ```

   Example Oracle Linux kernel version:

   ```
   6.12.0-1.23.3.2.el9uek.x86_64
   ```

2. Install the `nvme-cli` package:

   ```
   rpm -qa|grep nvme-cli
   ```

The following example shows an `nvme-cli` package version:

```
nvme-cli-2.11-5.el9.x86_64
```

3. Install the `libnvme` package:

```
rpm -qa|grep libnvme
```

The following example shows an `libnvme` package version:

```
libnvme-1.11.1-1.el9.x86_64
```

4. On the Oracle Linux 9.x host, check the `hostnqn` string at `/etc/nvme/hostnqn`:

```
cat /etc/nvme/hostnqn
```

The following example shows an `hostnqn` version:

```
nqn.2014-08.org.nvmexpress:uuid:b1d95cd0-1f7c-11ec-b8d1-3a68dd61a1cb
```

5. On the ONTAP system, verify that the `hostnqn` string matches the `hostnqn` string for the corresponding subsystem on the ONTAP storage system:

```
vserver nvme subsystem host show -vserver vs_203
```

Show example

```
Vserver Subsystem Priority  Host NQN
------- --------- --------
-------------------------------------------------------------------------
vs_203  Nvme1     regular   nqn.2014-
08.org.nvmexpress:uuid:b1d95cd0-1f7c-11ec-b8d1-3a68dd61a1cb
        Nvme10    regular   nqn.2014-
08.org.nvmexpress:uuid:b1d95cd0-1f7c-11ec-b8d1-3a68dd61a1cb
        Nvme11    regular   nqn.2014-
08.org.nvmexpress:uuid:b1d95cd0-1f7c-11ec-b8d1-3a68dd61a1cb
        Nvme12    regular   nqn.2014-
08.org.nvmexpress:uuid:b1d95cd0-1f7c-11ec-b8d1-3a68dd61a1cb
        Nvme13    regular   nqn.2014-
08.org.nvmexpress:uuid:b1d95cd0-1f7c-11ec-b8d1-3a68dd61a1cb
        Nvme14    regular   nqn.2014-
08.org.nvmexpress:uuid:b1d95cd0-1f7c-11ec-b8d1-3a68dd61a1cb
```

If the `hostnqn` strings don't match, you can use the `vserver modify` command to update the `hostnqn` string on your corresponding ONTAP array subsystem to match the `hostnqn` string from `/etc/nvme/hostnqn` on the host.

Step 3: Configure NVMe/FC and NVMe/TCP

Configure NVMe/FC with Broadcom/Emulex or Marvell/QLogic adapters, or configure NVMe/TCP using manual discovery and connect operations.

NVMe/FC - Broadcom/Emulex

Configure NVMe/FC for a Broadcom/Emulex adapter.

Steps

1. Verify that you're using the supported adapter model:
   a. Display the model names:

```
cat /sys/class/scsi_host/host*/modelname
```

   You should see the following output:

```
LPe36002-M64-D
LPe36002-M64-D
```

   b. Display the model descriptions:

```
cat /sys/class/scsi_host/host*/modeldesc
```

   You should see an output similar to the following example:

```
Emulex LPe36002-M64-D 2-Port 64Gb Fibre Channel Adapter
Emulex LPe36002-M64-D 2-Port 64Gb Fibre Channel Adapter
```

2. Verify that you are using the recommended Broadcom `lpfc` firmware and inbox driver:
   a. Display the firmware version:

```
cat /sys/class/scsi_host/host*/fwrev
```

   The following example shows firmware versions:

```
14.4.576.17, sli-4:6:d
14.4.576.17, sli-4:6:d
```

   b. Display the inbox driver version:

```
cat /sys/module/lpfc/version
```

   The following example shows a driver version:

```
0:14.4.0.8
```

For the current list of supported adapter driver and firmware versions, see the Interoperability Matrix Tool.

3. Verify that `lpfc_enable_fc4_type` is set to `3`:

```
cat /sys/module/lpfc/parameters/lpfc_enable_fc4_type
```

4. Verify that you can view your initiator ports:

```
cat /sys/class/fc_host/host*/<port_name>
```

The following example shows port identities:

```
0x2100f4c7aa9d7c5c
0x2100f4c7aa9d7c5d
```

5. Verify that your initiator ports are online:

```
cat /sys/class/fc_host/host*/port_state
```

You should see the following output:

```
Online
Online
```

6. Verify that the NVMe/FC initiator ports are enabled and that the target ports are visible:

```
cat /sys/class/scsi_host/host*/nvme_info
```

Show example

```
NVME Initiator Enabled
XRI Dist lpfc0 Total 6144 IO 5894 ELS 250
NVME LPORT lpfc0 WWPN x100000620b3c0869 WWNN x200000620b3c0869
DID x080e00 ONLINE
NVME RPORT       WWPN x2001d039eabac36f WWNN x2000d039eabac36f
DID x021401 TARGET DISCSRVC ONLINE
NVME RPORT       WWPN x20e2d039eabac36f WWNN x20e1d039eabac36f
DID x02141f TARGET DISCSRVC ONLINE
NVME RPORT       WWPN x2011d039eabac36f WWNN x2010d039eabac36f
DID x021429 TARGET DISCSRVC ONLINE
NVME RPORT       WWPN x2002d039eabac36f WWNN x2000d039eabac36f
DID x021003 TARGET DISCSRVC ONLINE
NVME RPORT       WWPN x20e4d039eabac36f WWNN x20e1d039eabac36f
DID x02100f TARGET DISCSRVC ONLINE
NVME RPORT       WWPN x2012d039eabac36f WWNN x2010d039eabac36f
DID x021015 TARGET DISCSRVC ONLINE

NVME Statistics
LS: Xmt 0000027ccf Cmpl 0000027cca Abort 00000014
LS XMIT: Err 00000005  CMPL: xb 00000014 Err 00000014
Total FCP Cmpl 00000000000613ff Issue 00000000000613fc OutIO
fffffffffffffffd
        abort 00000007 noxri 00000000 nondlp 00000000 qdepth
00000000 wqerr 00000000 err 00000000
FCP CMPL: xb 0000000a Err 0000000d

NVME Initiator Enabled
XRI Dist lpfc1 Total 6144 IO 5894 ELS 250
NVME LPORT lpfc1 WWPN x100000620b3c086a WWNN x200000620b3c086a
DID x080000 ONLINE
NVME RPORT       WWPN x2004d039eabac36f WWNN x2000d039eabac36f
DID x021501 TARGET DISCSRVC ONLINE
NVME RPORT       WWPN x20e3d039eabac36f WWNN x20e1d039eabac36f
DID x02150f TARGET DISCSRVC ONLINE
NVME RPORT       WWPN x2014d039eabac36f WWNN x2010d039eabac36f
DID x021515 TARGET DISCSRVC ONLINE
NVME RPORT       WWPN x2003d039eabac36f WWNN x2000d039eabac36f
DID x02110b TARGET DISCSRVC ONLINE
NVME RPORT       WWPN x20e5d039eabac36f WWNN x20e1d039eabac36f
DID x02111f TARGET DISCSRVC ONLINE
NVME RPORT       WWPN x2013d039eabac36f WWNN x2010d039eabac36f
DID x021129 TARGET DISCSRVC ONLINE

NVME Statistics
```

```
LS: Xmt 0000027ca3 Cmpl 0000027ca2 Abort 00000017
LS XMIT: Err 00000001  CMPL: xb 00000017 Err 00000017
Total FCP Cmpl 000000000006369d Issue 000000000006369a OutIO
fffffffffffffd
        abort 00000007 noxri 00000000 nondlp 00000011 qdepth
00000000 wqerr 00000000 err 00000000
FCP CMPL: xb 00000008 Err 0000000c
```

NVMe/FC - Marvell/QLogic

Configure NVMe/FC for a Marvell/QLogic adapter.

Steps

1. Verify that you are running the supported adapter driver and firmware versions:

   ```
   cat /sys/class/fc_host/host*/symbolic_name
   ```

   The follow example shows driver and firware versions:

   ```
   QLE2872 FW:v9.15.03 DVR:v10.02.09.300-k
   ```

2. Verify that `ql2xnvmeenable` is set. This enables the Marvell adapter to function as an NVMe/FC initiator:

   ```
   cat /sys/module/qla2xxx/parameters/ql2xnvmeenable
   ```

   The expected output is 1.

NVMe/TCP

The NVMe/TCP protocol doesn't support the auto-connect operation. Instead, you can discover the NVMe/TCP subsystems and namespaces by performing the NVMe/TCP `connect` or `connect-all` operations manually.

Steps

1. Verify that the initiator port can fetch the discovery log page data across the supported NVMe/TCP LIFs:

   ```
   nvme discover -t tcp -w host-traddr -a traddr
   ```

Show example

```
nvme discover -t tcp -w 192.168.30.10 -a 192.168.30.58
=====Discovery Log Entry 0======
trtype:  tcp
adrfam:  ipv4
subtype: current discovery subsystem
treq:    not specified
portid:  8
trsvcid: 8009
subnqn:  nqn.1992-
08.com.netapp:sn.064a9b19b3ee11f09dcad039eabac370:discovery
traddr:  192.168.31.99
eflags:  explicit discovery connections, duplicate discovery
information
sectype: none
=====Discovery Log Entry 1======
trtype:  tcp
adrfam:  ipv4
subtype: current discovery subsystem
treq:    not specified
portid:  6
trsvcid: 8009
subnqn:  nqn.1992-
08.com.netapp:sn.064a9b19b3ee11f09dcad039eabac370:discovery
traddr:  192.168.30.99
eflags:  explicit discovery connections, duplicate discovery
information
sectype: none
=====Discovery Log Entry 2======
trtype:  tcp
adrfam:  ipv4
subtype: current discovery subsystem
treq:    not specified
portid:  7
trsvcid: 8009
subnqn:  nqn.1992-
08.com.netapp:sn.064a9b19b3ee11f09dcad039eabac370:discovery
traddr:  192.168.31.98
eflags:  explicit discovery connections, duplicate discovery
information
sectype: none
=====Discovery Log Entry 3======
trtype:  tcp
adrfam:  ipv4
subtype: current discovery subsystem
```

```
treq:    not specified
portid:  5
trsvcid: 8009
subnqn:  nqn.1992-
08.com.netapp:sn.064a9b19b3ee11f09dcad039eabac370:discovery
traddr:  192.168.30.98
eflags:  explicit discovery connections, duplicate discovery
information
sectype: none
=====Discovery Log Entry 4======
trtype:  tcp
adrfam:  ipv4
subtype: nvme subsystem
treq:    not specified
portid:  8
trsvcid: 4420
subnqn:  nqn.1992-
08.com.netapp:sn.064a9b19b3ee11f09dcad039eabac370:subsystem.subs
ys_kvm
traddr:  192.168.31.99
eflags:  none
sectype: none
=====Discovery Log Entry 5======
trtype:  tcp
adrfam:  ipv4
subtype: nvme subsystem
treq:    not specified
portid:  6
trsvcid: 4420
subnqn:  nqn.1992-
08.com.netapp:sn.064a9b19b3ee11f09dcad039eabac370:subsystem.subs
ys_kvm
traddr:  192.168.30.99
eflags:  none
sectype: none
=====Discovery Log Entry 6======
trtype:  tcp
adrfam:  ipv4
subtype: nvme subsystem
treq:    not specified
portid:  7
trsvcid: 4420
subnqn:  nqn.1992-
08.com.netapp:sn.064a9b19b3ee11f09dcad039eabac370:subsystem.subs
ys_kvm
traddr:  192.168.31.98
```

```
eflags:  none
sectype: none
=====Discovery Log Entry 7======
trtype:  tcp
adrfam:  ipv4
subtype: nvme subsystem
treq:    not specified
portid:  5
trsvcid: 4420
subnqn:  nqn.1992-
08.com.netapp:sn.064a9b19b3ee11f09dcad039eabac370:subsystem.subs
ys_kvm
traddr:  192.168.30.98
eflags:  none
sectype: none
```

2. Verify that the other NVMe/TCP initiator-target LIF combinations can successfully fetch discovery log page data:

```
nvme discover -t tcp -w host-traddr -a traddr
```

Show example

```
nvme discover -t tcp -w 192.168.30.10 -a 192.168.30.58
nvme discover -t tcp -w 192.168.30.10 -a 192.168.30.59
nvme discover -t tcp -w 192.168.31.10 -a 192.168.31.58
nvme discover -t tcp -w 192.168.31.10 -a 192.168.31.59
```

3. Run the `nvme connect-all` command across all the supported NVMe/TCP initiator-target LIFs across the nodes:

```
nvme connect-all -t tcp -w host-traddr -a traddr
```

Show example

```
nvme connect-all -t tcp -w 192.168.30.10 -a 192.168.30.58
nvme connect-all -t tcp -w 192.168.30.10 -a 192.168.30.59
nvme connect-all -t tcp -w 192.168.31.10 -a 192.168.31.58
nvme connect-all -t tcp -w 192.168.31.10 -a 192.168.31.59
```

Beginning with Oracle Linux 9.4, the setting for the NVMe/TCP `ctrl_loss_tmo timeout` is automatically set to "off". As a result:

- There are no limits on the number of retries (indefinite retry).
- You don't need to manually configure a specific `ctrl_loss_tmo timeout` duration when using the `nvme connect` or `nvme connect-all` commands (option -l).
- The NVMe/TCP controllers don't experience timeouts in the event of a path failure and remain connected indefinitely.

Step 4: Optionally, modify the iopolicy in the udev rules

The Oracle Linux 9.x the host sets the default iopolicy for NVMe-oF to `round-robin`. Beginning with Oracle Linux 9.6, you can change the iopolicy to `queue-depth` by modifying the udev rules file.

Steps

1. Open the udev rules file in a text editor with root privileges:

   ```
   /usr/lib/udev/rules.d/71-nvmf-netapp.rules
   ```

   You should see the following output:

   ```
   vi /usr/lib/udev/rules.d/71-nvmf-netapp.rules
   ```

2. Find the line that sets iopolicy for the NetApp ONTAP Controller.

   The following example shows an example rule:

   ```
   ACTION=="add", SUBSYSTEM=="nvme-subsystem", ATTR{subsystype}=="nvm",
   ATTR{model}=="NetApp ONTAP Controller", ATTR{iopolicy}="round-robin"
   ```

3. Modify the rule so that `round-robin` becomes `queue-depth`:

   ```
   ACTION=="add", SUBSYSTEM=="nvme-subsystem", ATTR{subsystype}=="nvm",
   ATTR{model}=="NetApp ONTAP Controller", ATTR{iopolicy}="queue-depth"
   ```

4. Reload the udev rules and apply the changes:

   ```
   udevadm control --reload
   udevadm trigger --subsystem-match=nvme-subsystem
   ```

5. Verify the current iopolicy for your subsystem. Replace <subsystem>, for example, `nvme-subsys0`.

```
cat /sys/class/nvme-subsystem/<subsystem>/iopolicy
```

You should see the following output:

```
queue-depth.
```

> The new iopolicy applies automatically to matching NetApp ONTAP Controller devices. No reboot is required.

Step 5: Optionally, enable 1MB I/O for NVMe/FC

ONTAP reports a Max Data Transfer Size (MDTS) of 8 in the Identify Controller data. This means the maximum I/O request size can be up to 1MB. To issue I/O requests of size 1MB for a Broadcom NVMe/FC host, you should increase the `lpfc` value of the `lpfc_sg_seg_cnt` parameter to 256 from the default value of 64.

> These steps don't apply to Qlogic NVMe/FC hosts.

Steps

1. Set the `lpfc_sg_seg_cnt` parameter to 256:

   ```
   cat /etc/modprobe.d/lpfc.conf
   ```

   You should see an output similar to the following example:

   ```
   options lpfc lpfc_sg_seg_cnt=256
   ```

2. Run the `dracut -f` command, and reboot the host.
3. Verify that the value for `lpfc_sg_seg_cnt` is 256:

   ```
   cat /sys/module/lpfc/parameters/lpfc_sg_seg_cnt
   ```

Step 6: Verify NVMe boot services

Beginning with Oracle Linux 9.5, the `nvmefc-boot-connections.service` and `nvmf-autoconnect.service` boot services included in the NVMe/FC `nvme-cli` package are automatically enabled when the system boots.

After booting completes, verify that the `nvmefc-boot-connections.service` and `nvmf-autoconnect.service` boot services are enabled.

Steps

1. Verify that `nvmf-autoconnect.service` is enabled:

```
systemctl status nvmf-autoconnect.service
```

Show example output

```
nvmf-autoconnect.service - Connect NVMe-oF subsystems automatically
during boot
     Loaded: loaded (/usr/lib/systemd/system/nvmf-
autoconnect.service; enabled; preset: disabled)
     Active: inactive (dead) since Tue 2025-10-07 09:48:11 EDT; 1
week 0 days ago
   Main PID: 2620 (code=exited, status=0/SUCCESS)
        CPU: 19ms

Oct 07 09:48:11 R650xs-13-211 systemd[1]: Starting Connect NVMe-oF
subsystems automatically during boot...
Oct 07 09:48:11 R650xs-13-211 systemd[1]: nvmf-autoconnect.service:
Deactivated successfully.
Oct 07 09:48:11 R650xs-13-211 systemd[1]: Finished Connect NVMe-oF
subsystems automatically during boot.
```

2. Verify that `nvmefc-boot-connections.service` is enabled:

```
systemctl status nvmefc-boot-connections.service
```

Show example output

```
nvmefc-boot-connections.service - Auto-connect to subsystems on FC-
NVME devices found during boot
     Loaded: loaded (/usr/lib/systemd/system/nvmefc-boot-
connections.service; enabled; preset: enabled)
     Active: inactive (dead) since Tue 2025-10-07 09:47:07 EDT; 1
week 0 days ago
   Main PID: 1651 (code=exited, status=0/SUCCESS)
        CPU: 14ms

Oct 07 09:47:07 R650xs-13-211 systemd[1]: Starting Auto-connect to
subsystems on FC-NVME devices found during boot...
Oct 07 09:47:07 R650xs-13-211 systemd[1]: nvmefc-boot-
connections.service: Deactivated successfully.
Oct 07 09:47:07 R650xs-13-211 systemd[1]: Finished Auto-connect to
subsystems on FC-NVME devices found during boot.
```

Step 7: Verify the multipathing configuration

Verify that the in-kernel NVMe multipath status, ANA status, and ONTAP namespaces are correct for the NVMe-oF configuration.

Steps

1. Verify that the in-kernel NVMe multipath is enabled:

   ```
   cat /sys/module/nvme_core/parameters/multipath
   ```

   You should see the following output:

   ```
   Y
   ```

2. Verify that the appropriate NVMe-oF settings (such as, model set to NetApp ONTAP Controller and load balancing iopolicy set to round-robin) for the respective ONTAP namespaces correctly reflect on the host:

   a. Display the subsystems:

      ```
      cat /sys/class/nvme-subsystem/nvme-subsys*/model
      ```

      You should see the following output:

```
NetApp ONTAP Controller
NetApp ONTAP Controller
```

b. Display the policy:

```
cat /sys/class/nvme-subsystem/nvme-subsys*/iopolicy
```

You should see the value set for iopolicy, for example:

```
queue-depth
queue-depth
```

3. Verify that the namespaces are created and correctly discovered on the host:

```
nvme list
```

Show example

```
Node            Generic         SN                   Model
Namespace  Usage                      Format         FW Rev
--------------- --------------- --------------------
------------------------ ---------- --------------------------
-------------  --------
/dev/nvme102n1  /dev/ng102n1    81LLqNYTindCAAAAAAAk NetApp ONTAP
Controller   0x1          2.25  GB /   5.37  GB   4 KiB +  0 B
9.17.1
/dev/nvme102n2  /dev/ng102n2    81LLqNYTindCAAAAAAAk NetApp ONTAP
Controller   0x2          2.25  GB /   5.37  GB   4 KiB +  0 B
9.17.1
/dev/nvme106n1  /dev/ng106n1    81LLqNYTindCAAAAAAAs NetApp ONTAP
Controller   0x1          2.25  GB /   5.37  GB   4 KiB +  0 B
9.17.1
/dev/nvme106n2  /dev/ng106n2    81LLqNYTindCAAAAAAAs NetApp ONTAP
Controller   0x2          2.25  GB /   5.37  GB   4 KiB +  0 B
9.17.1
```

4. Verify that the controller state of each path is live and has the correct ANA status:

NVMe/FC

```
nvme list-subsys /dev/nvme4n5
```

Show example

```
nvme-subsys4 - NQN=nqn.1992-
08.com.netapp:sn.f9c6d0cb4fef11f08579d039eaa8138c:discovery
hostnqn=nqn.2014-08.org.nvmexpress:uuid:b1d95cd0-1f7c-11ec-b8d1-
3a68dd61a1cb \ +- nvme2 fc traddr=nn-0x201ad039eabac36f:pn-
0x201bd039eabac36f,host_traddr=nn-0x2000f4c7aa9d7c5c:pn-
0x2100f4c7aa9d7c5c live optimized
+- nvme8 fc traddr=nn-0x201ad039eabac36f:pn-
0x201dd039eabac36f,host_traddr=nn-0x2000f4c7aa9d7c5d:pn-
0x2100f4c7aa9d7c5d live non-optimized
+- nvme2 fc traddr=nn-0x201ad039eabac36f:pn-
0x201bd039eabac36f,host_traddr=nn-0x2000f4c7aa9d7c5c:pn-
0x2100f4c7aa9d7c5c live non-optimized
+- nvme8 fc traddr=nn-0x201ad039eabac36f:pn-
0x201dd039eabac36f,host_traddr=nn-0x2000f4c7aa9d7c5d:pn-
0x2100f4c7aa9d7c5d live optimized
```

NVMe/TCP

```
nvme list-subsys /dev/nvme1n1
```

Show example

```
nvme-subsys98 - NQN=nqn.1992-
08.com.netapp:sn.f9c6d0cb4fef11f08579d039eaa8138c:subsystem.Nvme
9
                hostnqn=nqn.2014-
08.org.nvmexpress:uuid:b1d95cd0-1f7c-11ec-b8d1-3a68dd61a1cb
\
 +- nvme100 fc traddr=nn-0x201ad039eabac36f:pn-
0x201dd039eabac36f,host_traddr=nn-0x2000f4c7aa9d7c5d:pn-
0x2100f4c7aa9d7c5d live non-optimized
 +- nvme101 fc traddr=nn-0x201ad039eabac36f:pn-
0x201cd039eabac36f,host_traddr=nn-0x2000f4c7aa9d7c5c:pn-
0x2100f4c7aa9d7c5c live non-optimized
 +- nvme98 fc traddr=nn-0x201ad039eabac36f:pn-
0x201bd039eabac36f,host_traddr=nn-0x2000f4c7aa9d7c5c:pn-
0x2100f4c7aa9d7c5c live optimized
 +- nvme99 fc traddr=nn-0x201ad039eabac36f:pn-
0x201ed039eabac36f,host_traddr=nn-0x2000f4c7aa9d7c5d:pn-
0x2100f4c7aa9d7c5d live optimized
[root@SR630-13-203 ~]#
```

5. Verify that the NetApp plug-in displays the correct values for each ONTAP namespace device:

Column

```
nvme netapp ontapdevices -o column
```

Show example

```
Device           Vserver    Namespace Path          NSID UUID
Size
---------------- ---------- ---------------------- ----
-------------------------------------- ------
/dev/nvme102n1   vs_203     /vol/Nvmevol35/ns35     1
00e760c9-e4ca-4d9f-b1d4-e9a930bf53c0   5.37GB
/dev/nvme102n2   vs_203     /vol/Nvmevol83/ns83     2
1fa97524-7dc2-4dbc-b4cf-5dda9e7095c0   5.37GB
```

JSON

```
nvme netapp ontapdevices -o json
```

Show example

```
{
  "ONTAPdevices":[
    {
      "Device":"/dev/nvme11n1",
      "Vserver":"vs_203",
      "Namespace_Path":"/vol/Nvmevol16/ns16",
      "NSID":1,
      "UUID":"18a88771-8b5b-4eb7-bff0-2ae261f488e4",
      "LBA_Size":4096,
      "Namespace_Size":5368709120,
      "UsedBytes":2262282240,
      "Version":"9.17.1"
        }
  ]
}
```

Step 8: Set up secure in-band authentication

Secure in-band authentication is supported over NVMe/TCP between a Oracle Linux 9.x host and an ONTAP

controller.

Each host or controller must be associated with a DH-HMAC-CHAP key to set up secure authentication. A DH-HMAC-CHAP key is a combination of the NQN of the NVMe host or controller and an authentication secret configured by the administrator. To authenticate its peer, an NVMe host or controller must recognize the key associated with the peer.

Steps

Set up secure in-band authentication using the CLI or a config JSON file. Use a config JSON file if you need to specify different dhchap keys for different subsystems.

CLI

Set up secure in-band authentication using the CLI.

1. Obtain the host NQN:

```
cat /etc/nvme/hostnqn
```

2. Generate the dhchap key for the Linux host.

   The following output describes the `gen-dhchap-key` command paramters:

```
nvme gen-dhchap-key -s optional_secret -l key_length {32|48|64} -m
HMAC_function {0|1|2|3} -n host_nqn
•   -s secret key in hexadecimal characters to be used to initialize
the host key
•   -l length of the resulting key in bytes
•   -m HMAC function to use for key transformation
0 = none, 1- SHA-256, 2 = SHA-384, 3=SHA-512
•   -n host NQN to use for key transformation
```

   In the following example, a random dhchap key with HMAC set to 3 (SHA-512) is generated.

```
# nvme gen-dhchap-key -m 3 -n nqn.2014-
08.org.nvmexpress:uuid:4c4c4544-0056-5410-8048-c4c04f425633
DHHC-
1:03:xhAfbAD5IVLZDxiVbmFEOA5JZ3F/ERqTXhHzZQJKgkYkTbPI9dhRyVtr4dBD+SG
iAJO3by4FbnVtov1Lmk+86+nNc6k=:
```

3. On the ONTAP controller, add the host and specify both dhchap keys:

```
vserver nvme subsystem host add -vserver <svm_name> -subsystem
<subsystem> -host-nqn <host_nqn> -dhchap-host-secret
<authentication_host_secret> -dhchap-controller-secret
<authentication_controller_secret> -dhchap-hash-function {sha-
256|sha-512} -dhchap-group {none|2048-bit|3072-bit|4096-bit|6144-
bit|8192-bit}
```

4. A host supports two types of authentication methods, unidirectional and bidirectional. On the host, connect to the ONTAP controller and specify dhchap keys based on the chosen authentication method:

```
nvme connect -t tcp -w <host-traddr> -a <tr-addr> -n <host_nqn> -S
<authentication_host_secret> -C <authentication_controller_secret>
```

5. Validate the `nvme connect authentication` command by verifying the host and controller dhchap keys:
   a. Verify the host dhchap keys:

```
cat /sys/class/nvme-subsystem/<nvme-subsysX>/nvme*/dhchap_secret
```

Show example output for a unidirectional configuration

```
cat /sys/class/nvme-subsystem/nvme-subsys1/nvme*/dhchap_secret
DHHC-
1:03:Y5VkkESgmtTGNdX842qemNpFK6BXYVwwnqErgt3IQKP5Fbjje\/JSBOjG
5Ea3NBLRfuiAuUSDUto6eY\/GwKoRp6AwGkw=:
DHHC-
1:03:Y5VkkESgmtTGNdX842qemNpFK6BXYVwwnqErgt3IQKP5Fbjje\/JSBOjG
5Ea3NBLRfuiAuUSDUto6eY\/GwKoRp6AwGkw=:
DHHC-
1:03:Y5VkkESgmtTGNdX842qemNpFK6BXYVwwnqErgt3IQKP5Fbjje\/JSBOjG
5Ea3NBLRfuiAuUSDUto6eY\/GwKoRp6AwGkw=:
DHHC-
1:03:Y5VkkESgmtTGNdX842qemNpFK6BXYVwwnqErgt3IQKP5Fbjje\/JSBOjG
5Ea3NBLRfuiAuUSDUto6eY\/GwKoRp6AwGkw=:
```

   b. Verify the controller dhchap keys:

```
cat /sys/class/nvme-subsystem/<nvme-
subsysX>/nvme*/dhchap_ctrl_secret
```

Show example output for a bidirectional configuration

```
cat /sys/class/nvme-subsystem/nvme-
subsys6/nvme*/dhchap_ctrl_secret
DHHC-
1:03:frpLlTrnOYtcWDxPzq4ccxU1UrH2FjV7hYw5s2XEDB+lo+TjMsOwHR\/N
FtM0nBBidx+gdoyUcC5s6hOOtTLDGcz0Kbs=:
DHHC-
1:03:frpLlTrnOYtcWDxPzq4ccxU1UrH2FjV7hYw5s2XEDB+lo+TjMsOwHR\/N
FtM0nBBidx+gdoyUcC5s6hOOtTLDGcz0Kbs=:
DHHC-
1:03:frpLlTrnOYtcWDxPzq4ccxU1UrH2FjV7hYw5s2XEDB+lo+TjMsOwHR\/N
FtM0nBBidx+gdoyUcC5s6hOOtTLDGcz0Kbs=:
DHHC-
1:03:frpLlTrnOYtcWDxPzq4ccxU1UrH2FjV7hYw5s2XEDB+lo+TjMsOwHR\/N
FtM0nBBidx+gdoyUcC5s6hOOtTLDGcz0Kbs=:
```

JSON

When multiple NVMe subsystems are available on the ONTAP controller configuration, you can use the `/etc/nvme/config.json` file with the `nvme connect-all` command.

Use the `-o` option to generate the JSON file. See the NVMe connect-all manual pages for more syntax options.

1. Configure the JSON file:

Show example

```
[
  {
    "hostnqn":"nqn.2014-08.org.nvmexpress:uuid:4c4c4544-0056-
5410-8048-c4c04f425633",
    "hostid":"4c4c4544-0056-5410-8048-c4c04f425633",
    "dhchap_key":"DHHC-
1:01:nFg06gV0FNpXqoiLOF0L+swULQpZU/PjU9v/McDeJHjTZFlF:",
    "subsystems":[
      {
        "nqn":"nqn.1992-
08.com.netapp:sn.09035a8d8c8011f0ac0fd039eabac370:subsystem.subs
ys",
        "ports":[
          {
            "transport":"tcp",
            "traddr":"192.168.30.69",
            "host_traddr":"192.168.30.10",
            "trsvcid":"4420",
            "dhchap_ctrl_key":"DHHC-
1:03:n3F8d+bvxKW/s+lEhqXaOohI2sxrQ9iLutzduuFq49JgdjjaFtTpDSO9kQl
/bvZj+Bo3rdHh3xPXeP6a4xyhcRyqdds=:"
          }
        ]
      }
    ]
  }
]
```

In the preceding example, `dhchap_key` corresponds to `dhchap_secret` and `dhchap_ctrl_key` corresponds to `dhchap_ctrl_secret`.

2. Connect to the ONTAP controller using the config JSON file:

```
nvme connect-all -J /etc/nvme/config.json
```

3. Verify that the dhchap secrets have been enabled for the respective controllers for each subsystem:
   a. Verify the host dhchap keys:

```
cat /sys/class/nvme-subsystem/nvme-subsys0/nvme0/dhchap_secret
```

The following example shows a dhchap key:

```
DHHC-1:01:nFg06gV0FNpXqoiLOF0L+swULQpZU/PjU9v/McDeJHjTZFlF:
```

b. Verify the controller dhchap keys:

```
cat /sys/class/nvme-subsystem/nvme-
subsys0/nvme0/dhchap_ctrl_secret
```

You should see an output similar to the following example:

```
DHHC-
1:03:n3F8d+bvxKW/s+lEhqXaOohI2sxrQ9iLutzduuFq49JgdjjaFtTpDSO9kQl/
bvZj+Bo3rdHh3xPXeP6a4xyhcRyqdds=:
```

Step 9: Review the known issues

There are no known issues.

Configure Oracle Linux 8.x with NVMe-oF for ONTAP storage

Oracle Linux hosts support the NVMe over Fibre Channel (NVMe/FC) and NVMe over TCP (NVMe/TCP) protocols with Asymmetric Namespace Access (ANA). ANA provides multipathing functionality equivalent to asymmetric logical unit access (ALUA) in iSCSI and FCP environments.

Learn how to configure NVMe over Fabrics (NVMe-oF) hosts for Oracle Linux 8.x. For more support and feature information, see Oracle Linux ONTAP support and features.

NVMe-oF with Oracle Linux 8.x has the following known limitations:

- SAN booting using the NVMe-oF protocol is not supported.
- NetApp sanlun host utility support isn't available for NVMe-oF on an Oracle Linux 8.x host. Instead, you can rely on the NetApp plug-in included in the native `nvme-cli` package for all NVMe-oF transports.
- For Oracle Linux 8.2 and earlier, native NVMe/FC auto-connect scripts are not available in the nvme-cli package. Use the HBA vendor provided external auto-connect scripts.
- For Oracle Linux 8.2 and earlier, round-robin load balancing is not enabled by default for NVMe multipathing. To enable this functionality, go to the step for writing a udev rule.

Step 1: Install Oracle Linux and NVMe software and verify your configuration

Use the following procedure to validate the minimum supported Oracle Linux 8.x software versions.

Steps

1. Install Oracle Linux 8.x on the server. After the installation is complete, verify that you are running the specified Oracle Linux 8.x kernel.

```
uname -r
```

   Example Oracle Linux kernel version:

```
5.15.0-206.153.7.1.el8uek.x86_64
```

2. Install the `nvme-cli` package:

```
rpm -qa|grep nvme-cli
```

   The following example shows an `nvme-cli` package version:

```
nvme-cli-1.16-9.el8.x86_64
```

3. For Oracle Linux 8.2 and earlier, add the following string as a separate udev rule for `/lib/udev/rules.d/71-nvme-iopolicy-netapp-ONTAP.rules`. This enables round-robin load balancing for NVMe multipath.

```
cat /lib/udev/rules.d/71-nvme-iopolicy-netapp-ONTAP.rules
Enable round-robin for NetApp ONTAP
ACTION=="add", SUBSYSTEMS=="nvme-subsystem", ATTRS{model}=="NetApp ONTAP
Controller", ATTR{iopolicy}="round-robin"
```

4. On the Oracle Linux 8.x host, check the `hostnqn` string at `/etc/nvme/hostnqn`:

```
cat /etc/nvme/hostnqn
```

   The following example shows an `hostnqn` version:

```
nqn.2014-08.org.nvmexpress:uuid:edd38060-00f7-47aa-a9dc-4d8ae0cd969a
```

5. On the ONTAP system, verify that the `hostnqn` string matches the `hostnqn` string for the corresponding subsystem on the ONTAP storage system:

```
vserver nvme subsystem host show -vserver vs_coexistence_LPE36002
```

Show example

```
Vserver Subsystem Priority  Host NQN
------- --------- --------
-------------------------------------------------
vs_coexistence_LPE36002
        nvme
                  regular   nqn.2014-
08.org.nvmexpress:uuid:edd38060-00f7-47aa-a9dc-4d8ae0cd969a
        nvme1
                  regular   nqn.2014-
08.org.nvmexpress:uuid:edd38060-00f7-47aa-a9dc-4d8ae0cd969a
        nvme2
                  regular   nqn.2014-
08.org.nvmexpress:uuid:edd38060-00f7-47aa-a9dc-4d8ae0cd969a
        nvme3
                  regular   nqn.2014-
08.org.nvmexpress:uuid:edd38060-00f7-47aa-a9dc-4d8ae0cd969a
4 entries were displayed.
```

If the `hostnqn` strings don't match, use the `vserver modify` command to update the `hostnqn` string on your corresponding ONTAP array subsystem to match the `hostnqn` string from `/etc/nvme/hostnqn` on the host.

6. Optionally, to run both NVMe and SCSI co-existent traffic on the same host, NetApp recommends using the in-kernel NVMe multipath for ONTAP namespaces and `dm-multipath` for ONTAP LUNs respectively. This should exclude the ONTAP namespaces from `dm-multipath` and prevent `dm-multipath` from claiming the ONTAP namespace devices.

   a. Add the `enable_foreign` setting to the `/etc/multipath.conf` file.

   ```
   cat /etc/multipath.conf
   defaults {
     enable_foreign     NONE
   }
   ```

   b. Restart the `multipathd` daemon to apply the new setting.

   ```
   systemctl restart multipathd
   ```

Step 2: Configure NVMe/FC and NVMe/TCP

Configure NVMe/FC with Broadcom/Emulex or Marvell/QLogic adapters, or configure NVMe/TCP using

manual discovery and connect operations.

FC - Broadcom/Emulex

Configure NVMe/FC for a Broadcom/Emulex adapter.

Steps

1. Verify that you're using the supported adapter model:
   a. Display the model names:

```
cat /sys/class/scsi_host/host*/modelname
```

   You should see the following output:

```
LPe36002-M64
LPe36002-M64
```

   b. Display the model descriptions:

```
cat /sys/class/scsi_host/host*/modeldesc
```

   You should see an output similar to the following example:

```
Emulex LPe36002-M64 2-Port 64Gb Fibre Channel Adapter
Emulex LPe36002-M64 2-Port 64Gb Fibre Channel Adapter
```

2. Verify that you are using the recommended Broadcom `lpfc` firmware and inbox driver:
   a. Display the firmware version:

```
cat /sys/class/scsi_host/host*/fwrev
```

   The following example shows firmware versions:

```
14.4.317.10, sli-4:6:d
14.4.317.10, sli-4:6:d
```

   b. Display the inbox driver version:

```
cat /sys/module/lpfc/version
```

   The following example shows a driver version:

```
0:14.2.0.13
```

For the current list of supported adapter driver and firmware versions, see the Interoperability Matrix Tool.

3. Verify that `lpfc_enable_fc4_type` is set to "3":

```
cat /sys/module/lpfc/parameters/lpfc_enable_fc4_type
```

4. Verify that you can view your initiator ports:

```
cat /sys/class/fc_host/host*/<port_name>
```

The following example shows port identities:

```
0x100000109bf0449c
0x100000109bf0449d
```

5. Verify that your initiator ports are online:

```
cat /sys/class/fc_host/host*/port_state
```

You should see the following output:

```
Online
Online
```

6. Verify that the NVMe/FC initiator ports are enabled and that the target ports are visible:

```
cat /sys/class/scsi_host/host*/nvme_info
```

Show example

```
NVME Initiator Enabled
XRI Dist lpfc0 Total 6144 IO 5894 ELS 250
NVME LPORT lpfc0 WWPN x100000109bf0449c WWNN x200000109bf0449c
DID x061500 ONLINE
NVME RPORT       WWPN x200bd039eab31e9c WWNN x2005d039eab31e9c
DID x020e06 TARGET DISCSRVC ONLINE
NVME RPORT       WWPN x2006d039eab31e9c WWNN x2005d039eab31e9c
DID x020a0a TARGET DISCSRVC ONLINE
NVME Statistics
LS: Xmt 000000002c Cmpl 000000002c Abort 00000000
LS XMIT: Err 00000000  CMPL: xb 00000000 Err 00000000
Total FCP Cmpl 000000000008ffe8 Issue 000000000008ffb9 OutIO
fffffffffffffd1
        abort 0000000c noxri 00000000 nondlp 00000000 qdepth
00000000 wqerr 00000000 err 00000000
FCP CMPL: xb 0000000c Err 0000000c
NVME Initiator Enabled
XRI Dist lpfc1 Total 6144 IO 5894 ELS 250
NVME LPORT lpfc1 WWPN x100000109bf0449d WWNN x200000109bf0449d
DID x062d00 ONLINE
NVME RPORT       WWPN x201fd039eab31e9c WWNN x2005d039eab31e9c
DID x02090a TARGET DISCSRVC ONLINE
NVME RPORT       WWPN x200cd039eab31e9c WWNN x2005d039eab31e9c
DID x020d06 TARGET DISCSRVC ONLINE
NVME Statistics
LS: Xmt 0000000041 Cmpl 0000000041 Abort 00000000
LS XMIT: Err 00000000  CMPL: xb 00000000 Err 00000000
Total FCP Cmpl 00000000000936bf Issue 000000000009369a OutIO
fffffffffffffdb
        abort 00000016 noxri 00000000 nondlp 00000000 qdepth
00000000 wqerr 00000000 err 00000000
FCP CMPL: xb 00000016 Err 00000016
```

FC - Marvell/QLogic

Configure NVMe/FC for a Marvell/QLogic adapter.

Steps

1. Verify that you are running the supported adapter driver and firmware versions:

```
cat /sys/class/fc_host/host*/symbolic_name
```

The follow example shows driver and firware versions:

```
QLE2772 FW:v9.15.00 DVR:v10.02.09.100-k
QLE2772 FW:v9.15.00 DVR:v10.02.09.100-k
```

2. Verify that `ql2xnvmeenable` is set. This enables the Marvell adapter to function as an NVMe/FC initiator:

```
cat /sys/module/qla2xxx/parameters/ql2xnvmeenable
```

The expected output is 1.

TCP

The NVMe/TCP protocol doesn't support the auto-connect operation. Instead, you can discover the NVMe/TCP subsystems and namespaces by performing the NVMe/TCP `connect` or `connect-all` operations manually.

1. Verify that the initiator port can fetch the discovery log page data across the supported NVMe/TCP LIFs:

```
nvme discover -t tcp -w <host-traddr> -a <traddr>
```

Show example

```
nvme discover -t tcp -w 192.168.6.1 -a 192.168.6.24 Discovery
Log Number of Records 20, Generation counter 45
=====Discovery Log Entry 0======
trtype:  tcp
adrfam:  ipv4
subtype: unrecognized
treq:    not specified
portid:  6
trsvcid: 8009
subnqn:  nqn.1992-
08.com.netapp:sn.e6c438e66ac211ef9ab8d039eab31e9d:discovery
traddr:  192.168.6.25
sectype: none
=====Discovery Log Entry 1======
trtype:  tcp
adrfam:  ipv4
subtype: unrecognized
treq:    not specified
portid:  1
trsvcid: 8009
subnqn:  nqn.1992-
08.com.netapp:sn.e6c438e66ac211ef9ab8d039eab31e9d:discovery
traddr:  192.168.5.24
sectype: none
=====Discovery Log Entry 2======
trtype:  tcp
adrfam:  ipv4
subtype: unrecognized
treq:    not specified
portid:  4
trsvcid: 8009
subnqn:  nqn.1992-
08.com.netapp:sn.e6c438e66ac211ef9ab8d039eab31e9d:discovery
traddr:  192.168.6.24
sectype: none
=====Discovery Log Entry 3======
trtype:  tcp
adrfam:  ipv4
subtype: unrecognized
treq:    not specified
portid:  2
trsvcid: 8009
subnqn:  nqn.1992-
08.com.netapp:sn.e6c438e66ac211ef9ab8d039eab31e9d:discovery
```

```
traddr:  192.168.5.25
sectype: none
=====Discovery Log Entry 4======
trtype:  tcp
adrfam:  ipv4
subtype: nvme subsystem
treq:    not specified
portid:  6
trsvcid: 4420
subnqn:  nqn.1992-
08.com.netapp:sn.e6c438e66ac211ef9ab8d039eab31e9d:subsystem.nvme
_tcp_4
traddr:  192.168.6.25
sectype: none
=====Discovery Log Entry 5======
trtype:  tcp
adrfam:  ipv4
subtype: nvme subsystem
treq:    not specified
portid:  1
trsvcid: 4420
subnqn:  nqn.1992-
08.com.netapp:sn.e6c438e66ac211ef9ab8d039eab31e9d:subsystem.nvme
_tcp_4
..........
```

2. Verify that all other NVMe/TCP initiator-target LIF combinations can successfully fetch discovery log page data:

```
nvme discover -t tcp -w <host-traddr> -a <traddr>
```

Show example

```
nvme discover -t tcp -w 192.168.6.1 -a 192.168.6.24
nvme discover -t tcp -w 192.168.6.1 -a 192.168.6.25
nvme discover -t tcp -w 192.168.5.1 -a 192.168.5.24
nvme discover -t tcp -w 192.168.5.1 -a 192.168.5.25
```

3. Run the `nvme connect-all` command across all the supported NVMe/TCP initiator-target LIFs across the nodes:

```
nvme connect-all -t tcp -w host-traddr -a traddr -l
<ctrl_loss_timeout_in_seconds>
```

Show example

```
nvme    connect-all -t  tcp -w  192.168.5.1 -a  192.168.5.24
-l -1
nvme    connect-all -t  tcp -w  192.168.5.1 -a  192.168.5.25
-l -1
nvme    connect-all -t  tcp -w  192.168.6.1 -a  192.168.6.24
-l -1
nvme    connect-all -t  tcp -w  192.168.6.1 -a  192.168.6.25
-l -1
```

NetApp recommends setting the `ctrl-loss-tmo option` to `-1` so that the NVMe/TCP initiator attempts to reconnect indefinitely in the event of a path loss.

Step 3: Optionally, enable 1MB I/O for NVMe/FC

ONTAP reports a Max Data Transfer Size (MDTS) of 8 in the Identify Controller data. This means the maximum I/O request size can be up to 1MB. To issue I/O requests of size 1MB for a Broadcom NVMe/FC host, you should increase the `lpfc` value of the `lpfc_sg_seg_cnt` parameter to 256 from the default value of 64.

These steps don't apply to Qlogic NVMe/FC hosts.

Steps

1. Set the `lpfc_sg_seg_cnt` parameter to 256:

   ```
   cat /etc/modprobe.d/lpfc.conf
   ```

   You should see an output similar to the following example:

   ```
   options lpfc lpfc_sg_seg_cnt=256
   ```

2. Run the `dracut -f` command, and reboot the host.
3. Verify that the value for `lpfc_sg_seg_cnt` is 256:

   ```
   cat /sys/module/lpfc/parameters/lpfc_sg_seg_cnt
   ```

Step 4: Verify the multipathing configuration

Verify that the in-kernel NVMe multipath status, ANA status, and ONTAP namespaces are correct for the NVMe-oF configuration.

Steps

1. Verify that the in-kernel NVMe multipath is enabled:

   ```
   cat /sys/module/nvme_core/parameters/multipath
   ```

   You should see the following output:

   ```
   Y
   ```

2. Verify that the appropriate NVMe-oF settings (such as, model set to NetApp ONTAP Controller and load balancing iopolicy set to round-robin) for the respective ONTAP namespaces correctly reflect on the host:

   a. Display the subsystems:

      ```
      cat /sys/class/nvme-subsystem/nvme-subsys*/model
      ```

      You should see the following output:

      ```
      NetApp ONTAP Controller
      NetApp ONTAP Controller
      ```

   b. Display the policy:

      ```
      cat /sys/class/nvme-subsystem/nvme-subsys*/iopolicy
      ```

      You should see the following output:

      ```
      round-robin
      round-robin
      ```

3. Verify that the namespaces are created and correctly discovered on the host:

   ```
   nvme list
   ```

Show example

```
Node         SN                   Model
------------------------------------------------------------
/dev/nvme0n1 814vWBNRwf9HAAAAAAAB NetApp ONTAP Controller
/dev/nvme0n2 814vWBNRwf9HAAAAAAAB NetApp ONTAP Controller
/dev/nvme0n3 814vWBNRwf9HAAAAAAAB NetApp ONTAP Controller

Namespace Usage   Format               FW            Rev
------------------------------------------------------------
1                 85.90 GB / 85.90 GB  4 KiB + 0 B   FFFFFFFF
2                 85.90 GB / 85.90 GB  24 KiB + 0 B  FFFFFFFF
3                   85.90 GB / 85.90 GB  4 KiB + 0 B   FFFFFFFF
```

4. Verify that the controller state of each path is live and has the correct ANA status:

```
nvme list-subsys /dev/nvme0n1
```

Show NVMe/FC example

```
nvme-subsys0 - NQN=nqn.1992- 08.com.netapp:
4b4d82566aab11ef9ab8d039eab31e9d:subsystem.nvme\
+-  nvme1 fc traddr=nn-0x2038d039eab31e9c:pn-0x203ad039eab31e9c
host_traddr=nn-0x200034800d756a89:pn-0x210034800d756a89 live
optimized
+-  nvme2 fc traddr=nn-0x2038d039eab31e9c:pn-0x203cd039eab31e9c
host_traddr=nn-0x200034800d756a88:pn-0x210034800d756a88 live
optimized
+- nvme3 fc traddr=nn-0x2038d039eab31e9c:pn-0x203ed039eab31e9c
host_traddr=nn-0x200034800d756a89:pn-0x210034800d756a89 live non-
optimized
+-  nvme7 fc traddr=nn-0x2038d039eab31e9c:pn-0x2039d039eab31e9c
host_traddr=nn-0x200034800d756a88:pn-0x210034800d756a88 live non-
optimized
```

Show NVMe/TCP example

```
nvme-subsys0 - NQN=nqn.1992- 08.com.netapp:
sn.e6c438e66ac211ef9ab8d039eab31e9d:subsystem.nvme_tcp_4
\
+- nvme1 tcp traddr=192.168.5.25 trsvcid=4420
host_traddr=192.168.5.1 src_addr=192.168.5.1 live optimized
+- nvme10 tcp traddr=192.168.6.24 trsvcid=4420
host_traddr=192.168.6.1 src_addr=192.168.6.1 live optimized
+- nvme2 tcp traddr=192.168.5.24 trsvcid=4420
host_traddr=192.168.5.1 src_addr=192.168.5.1 live non-optimized
+- nvme9 tcp traddr=192.168.6.25 trsvcid=4420
host_traddr=192.168.6.1 src_addr=192.168.6.1 live non-optimized
```

5. Verify that the NetApp plug-in displays the correct values for each ONTAP namespace device:

Column

```
nvme netapp ontapdevices -o column
```

Show example

```
Device         Vserver                   Namespace Path
NSID UUID                                   Size
-------------- ------------------------
------------------------------ ----
-------------------------------------- ---------
/dev/nvme0n1   vs_coexistence_QLE2772
/vol/fcnvme_1_1_0/fcnvme_ns   1    159f9f88-be00-4828-aef6-
197d289d4bd9  10.74GB
/dev/nvme0n2   vs_coexistence_QLE2772
/vol/fcnvme_1_1_1/fcnvme_ns   2    2c1ef769-10c0-497d-86d7-
e84811ed2df6  10.74GB
/dev/nvme0n3   vs_coexistence_QLE2772
/vol/fcnvme_1_1_2/fcnvme_ns   3    9b49bf1a-8a08-4fa8-baf0-
6ec6332ad5a4  10.74GB
```

JSON

```
nvme netapp ontapdevices -o json
```

Show example

```
{
  "ONTAPdevices" : [
    {
      "Device" : "/dev/nvme0n1",
      "Vserver" : "vs_coexistence_QLE2772",
      "Namespace_Path" : "/vol/fcnvme_1_1_0/fcnvme_ns",
      "NSID" : 1,
      "UUID" : "159f9f88-be00-4828-aef6-197d289d4bd9",
      "Size" : "10.74GB",
      "LBA_Data_Size" : 4096,
      "Namespace_Size" : 2621440
    },
    {
      "Device" : "/dev/nvme0n2",
      "Vserver" : "vs_coexistence_QLE2772",
      "Namespace_Path" : "/vol/fcnvme_1_1_1/fcnvme_ns",
      "NSID" : 2,
      "UUID" : "2c1ef769-10c0-497d-86d7-e84811ed2df6",
      "Size" : "10.74GB",
      "LBA_Data_Size" : 4096,
      "Namespace_Size" : 2621440
    },
    {
      "Device" : "/dev/nvme0n4",
      "Vserver" : "vs_coexistence_QLE2772",
      "Namespace_Path" : "/vol/fcnvme_1_1_3/fcnvme_ns",
      "NSID" : 4,
      "UUID" : "f3572189-2968-41bc-972a-9ee442dfaed7",
      "Size" : "10.74GB",
      "LBA_Data_Size" : 4096,
      "Namespace_Size" : 2621440
    },
```

Step 5: Optionally, enable 1MB I/O size

ONTAP reports a Max Data Transfer Size (MDTS) of 8 in the Identify Controller data. This means the maximum I/O request size can be up to 1MB. To issue I/O requests of size 1MB for a Broadcom NVMe/FC host, you should increase the `lpfc` value of the `lpfc_sg_seg_cnt` parameter to 256 from the default value of 64.

These steps don't apply to Qlogic NVMe/FC hosts.

Steps

1. Set the `lpfc_sg_seg_cnt` parameter to 256:

```
cat /etc/modprobe.d/lpfc.conf
```

You should see an output similar to the following example:

```
options lpfc lpfc_sg_seg_cnt=256
```

2. Run the `dracut -f` command, and reboot the host.
3. Verify that the value for `lpfc_sg_seg_cnt` is 256:

```
cat /sys/module/lpfc/parameters/lpfc_sg_seg_cnt
```

Step 6: Review the known issues

These are the known issues:

NetApp Bug ID	Title	Description
1479047	Oracle Linux 8.x NVMe-oF hosts create duplicate Persistent Discovery Controllers (PDCs)	On NVMe-oF hosts, you can use the `nvme discover -p` command to create PDCs. When this command is used, only one PDC should be created per initiator-target combination. However, if you are running Oracle Linux 8.x with an NVMe-oF host, a duplicate PDC is created each time `nvme discover -p` is executed. This leads to unnecessary usage of resources on both the host and the target.

Configure Oracle Linux 7.x with NVMe-oF for ONTAP storage

Oracle Linux hosts support the NVMe over Fibre Channel (NVMe/FC) and NVMe over TCP (NVMe/TCP) protocols with Asymmetric Namespace Access (ANA). ANA provides multipathing functionality equivalent to asymmetric logical unit access (ALUA) in iSCSI and FCP environments.

Learn how to configure NVMe over Fabrics (NVMe-oF) hosts for Oracle Linux 7.x. For more support and feature information, see Oracle Linux ONTAP support and features.

NVMe-oF with Oracle Linux 7.x has the following known limitations:

- SAN booting using the NVMe-oF protocol is not supported.
- NetApp sanlun host utility support isn't available for NVMe-oF on an Oracle Linux 7.x host. Instead, you can rely on the NetApp plug-in included in the native `nvme-cli` package for all NVMe-oF transports.

- Native NVMe/FC auto-connect scripts are not available in the nvme-cli package. Use the HBA vendor provided external auto-connect scripts.
- Round-robin load balancing is not enabled by default for NVMe multipathing. To enable this functionality, write a udev rule.

Step 1: Install Oracle Linux and NVMe software and verify your configuration

Use the following procedure to validate the minimum supported Oracle Linux 7.x software versions.

Steps

1. Install Oracle Linux 7.x on the server. After the installation is complete, verify that you are running the specified Oracle Linux 7.x kernel.

   ```
   uname -r
   ```

   Example Oracle Linux kernel version:

   ```
   5.4.17-2011.6.2.el7uek.x86_64
   ```

2. Install the `nvme-cli` package:

   ```
   rpm -qa | grep nvme-cli
   ```

   The following example shows an `nvme-cli` package version:

   ```
   nvme-cli-1.8.1-3.el7.x86_64
   ```

3. Add the following string as a separate udev rule for `/lib/udev/rules.d/71-nvme-iopolicy-netapp-ONTAP.rules`. This enables round-robin load balancing for NVMe multipath.

   ```
   cat /lib/udev/rules.d/71-nvme-iopolicy-netapp-ONTAP.rules
   Enable round-robin for NetApp ONTAP
   ACTION=="add", SUBSYSTEMS=="nvme-subsystem", ATTRS{model}=="NetApp ONTAP
   Controller", ATTR{iopolicy}="round-robin"
   ```

4. On the Oracle Linux 7.x host, check the `hostnqn` string at `/etc/nvme/hostnqn`:

   ```
   cat /etc/nvme/hostnqn
   ```

   The following example shows an `hostnqn` version:

```
nqn.2014-08.org.nvmexpress:uuid:497ad959-e6d0-4987-8dc2-a89267400874
```

5. On the ONTAP system, verify that the `hostnqn` string matches the `hostnqn` string for the corresponding subsystem on the ONTAP storage system:

```
*> vserver nvme subsystem host show -vserver vs_nvme_10
```

Show example

```
Vserver Subsystem Host NQN
------- --------- ------------------------------------ -----------
ol_157_nvme_ss_10_0
nqn.2014-08.org.nvmexpress:uuid:497ad959-e6d0-4987-8dc2-a89267400874
```

If the `hostnqn` strings don't match, use the `vserver modify` command to update the `hostnqn` string on your corresponding ONTAP array subsystem to match the `hostnqn` string from `/etc/nvme/hostnqn` on the host.

6. Reboot the host.

Step 2: Configure NVMe/FC

Configure NVMe/FC for a Broadcom/Emulex adapter.

1. Verify that you're using the supported adapter model:
   a. Display the model names:

```
cat /sys/class/scsi_host/host*/modelname
```

   You should see the following output:

```
LPe32002-M2
LPe32002-M2
```

   b. Display the model descriptions:

```
cat /sys/class/scsi_host/host*/modeldesc
```

   You should see an output similar to the following example:

```
Emulex LightPulse LPe32002-M2 2-Port 32Gb Fibre Channel Adapter
Emulex LightPulse LPe32002-M2 2-Port 32Gb Fibre Channel Adapter
```

2. Verify that `lpfc_enable_fc4_type` is set to "3":

```
cat /sys/module/lpfc/parameters/lpfc_enable_fc4_type
```

3. Install the recommended lpfc auto-connect scripts:

```
rpm -ivh nvmefc-connect-12.8.264.0-1.noarch.rpm
```

4. Verify that the auto-connect scripts are installed:

```
rpm -qa | grep nvmefc
```

You should see the following output:

```
nvmefc-connect-12.8.264.0-1.noarch
```

5. Verify that your initiator ports are online:
   a. Display the port name:

```
cat /sys/class/fc_host/host*/port_name
```

You should see the following output:

```
0x10000090fae0ec61
0x10000090fae0ec62
```

   b. Display the port name:

```
cat /sys/class/fc_host/host*/port_state
```

You should see the following output:

```
Online
Online
```

6. Verify that the NVMe/FC initiator ports are enabled and that the target ports are visible:

```
cat /sys/class/scsi_host/host*/nvme_info
```

Show example

```
NVME Initiator Enabled
XRI Dist lpfc0 Total 6144 NVME 2947 SCSI 2947 ELS 250
NVME LPORT lpfc0 WWPN x10000090fae0ec61 WWNN x20000090fae0ec61 DID
x012000 ONLINE
NVME RPORT WWPN x202d00a098c80f09 WWNN x202c00a098c80f09 DID x010201
TARGET DISCSRVC ONLINE
NVME RPORT WWPN x203100a098c80f09 WWNN x202c00a098c80f09 DID x010601
TARGET DISCSRVC ONLINE
```

Step 3: Optionally, enable 1MB I/O for NVMe/FC

ONTAP reports a Max Data Transfer Size (MDTS) of 8 in the Identify Controller data. This means the maximum I/O request size can be up to 1MB. To issue I/O requests of size 1MB for a Broadcom NVMe/FC host, you should increase the `lpfc` value of the `lpfc_sg_seg_cnt` parameter to 256 from the default value of 64.

These steps don't apply to Qlogic NVMe/FC hosts.

Steps

1. Set the `lpfc_sg_seg_cnt` parameter to 256:

```
cat /etc/modprobe.d/lpfc.conf
```

You should see an output similar to the following example:

```
options lpfc lpfc_sg_seg_cnt=256
```

2. Run the `dracut -f` command, and reboot the host.
3. Verify that the value for `lpfc_sg_seg_cnt` is 256:

```
cat /sys/module/lpfc/parameters/lpfc_sg_seg_cnt
```

Step 4: Verify the multipathing configuration

Verify that the in-kernel NVMe multipath status, ANA status, and ONTAP namespaces are correct for the

NVMe-oF configuration.

Steps

1. Verify that the in-kernel NVMe multipath is enabled:

   ```
   cat /sys/module/nvme_core/parameters/multipath
   ```

   You should see the following output:

   ```
   Y
   ```

2. Verify that the appropriate NVMe-oF settings (such as, model set to NetApp ONTAP Controller and load balancing iopolicy set to round-robin) for the respective ONTAP namespaces correctly reflect on the host:

   a. Display the subsystems:

      ```
      cat /sys/class/nvme-subsystem/nvme-subsys*/model
      ```

      You should see the following output:

      ```
      NetApp ONTAP Controller
      NetApp ONTAP Controller
      ```

   b. Display the policy:

      ```
      cat /sys/class/nvme-subsystem/nvme-subsys*/iopolicy
      ```

      You should see the following output:

      ```
      round-robin
      round-robin
      ```

3. Verify that the namespaces are created and correctly discovered on the host:

   ```
   nvme list
   ```

Show example

```
Node SN Model Namespace Usage Format FW Rev
---------------- -------------------- ----------------------
/dev/nvme0n1 80BADBKnB/JvAAAAAAAC NetApp ONTAP Controller 1 53.69 GB
/ 53.69 GB 4 KiB + 0 B FFFFFFFF
```

4. Verify that the controller state of each path is live and has the correct ANA status:

```
nvme list-subsys /dev/nvme0n1
```

Show example

```
Nvme-subsysf0 - NQN=nqn.1992-
08.com.netapp:sn.341541339b9511e8a9b500a098c80f09:subsystem.ol_157_n
vme_ss_10_0
\
+- nvme0 fc traddr=nn-0x202c00a098c80f09:pn-0x202d00a098c80f09
host_traddr=nn-0x20000090fae0ec61:pn-0x10000090fae0ec61 live
optimized
+- nvme1 fc traddr=nn-0x207300a098dfdd91:pn-0x207600a098dfdd91
host_traddr=nn-0x200000109b1c1204:pn-0x100000109b1c1204 live
inaccessible
+- nvme2 fc traddr=nn-0x207300a098dfdd91:pn-0x207500a098dfdd91
host_traddr=nn-0x200000109b1c1205:pn-0x100000109b1c1205 live
optimized
+- nvme3 fc traddr=nn-0x207300a098dfdd91:pn-0x207700a098dfdd91 host
traddr=nn-0x200000109b1c1205:pn-0x100000109b1c1205 live inaccessible
```

5. Verify that the NetApp plug-in displays the correct values for each ONTAP namespace device:

Column

```
nvme netapp ontapdevices -o column
```

Show example

```
Device   Vserver  Namespace Path             NSID   UUID   Size
-------  -------- -------------------------  ------ ----- -----
/dev/nvme0n1   vs_nvme_10
/vol/rhel_141_vol_10_0/ol_157_ns_10_0    1        55baf453-f629-
4a18-9364-b6aee3f50dad   53.69GB
```

JSON

```
nvme netapp ontapdevices -o json
```

Show example

```
{
   "ONTAPdevices" : [
   {
        Device" : "/dev/nvme0n1",
        "Vserver" : "vs_nvme_10",
        "Namespace_Path" :
"/vol/rhel_141_vol_10_0/ol_157_ns_10_0",
         "NSID" : 1,
         "UUID" : "55baf453-f629-4a18-9364-b6aee3f50dad",
         "Size" : "53.69GB",
         "LBA_Data_Size" : 4096,
         "Namespace_Size" : 13107200
    }
]
```

Step 5: Review the known issues

There are no known issues.

Proxmox

Learn about ONTAP support and features for Proxmox hosts

Features supported for host configuration with NVMe over Fabrics (NVMe-oF) varies based on your version of ONTAP and Proxmox.

ONTAP feature	Proxmox host version	ONTAP version
NVMe/TCP is a fully supported enterprise feature	9.0	9.10.1 or later
Provides ONTAP details for both NVMe/FC and NVMe/TCP namespaces using the native `nvme-cli` package.	8.0	9.10.1 or later
NVMe and SCSI traffic is supported on the same host using NVMe multipath for NVMe-oF namespaces and dm-multipath for SCSI LUNs	8.0	9.4 or later

ONTAP supports the following SAN host features regardless of the ONTAP version running on your system setup.

Feature	Proxmox host version
The `nvme-cli` package includes auto-connect scripts removing the need for third-party scripts	9.0
The native udev rule in the `nvme-cli` package provides round-robin load balancing for NVMe multipathing	9.0
Native NVMe multipathing is enabled by default	8.0

For details on supported configurations, see the Interoperability Matrix Tool.

What's next

If your Proxmox VE version is ..	Learn about ..
9 series	Configuring NVMe for Proxmox VE 9.x
8 series	Configuring NVMe for Proxmox VE 8.x

Related information

Learn about managing NVMe protocols

Configure Proxmox VE 9.x for NVMe-oF with ONTAP storage

The Proxmox VE 9.x host supports the NVMe over Fibre Channel (NVMe/FC) and NVMe over TCP (NVMe/TCP) protocols with Asymmetric Namespace Access (ANA). ANA provides multipathing functionality equivalent to asymmetric logical unit access (ALUA) in iSCSI and FCP environments.

Learn how to configure NVMe over Fabrics (NVMe-oF) hosts for Proxmox VE 9.x. For more support and

feature information, see ONTAP support and features.

NVMe-oF with Proxmox VE 9.x has the following known limitation:

- The SAN boot configuration for NVMe-FC is not supported.

Step 1: Install Proxmox VE and NVMe software and verify your configuration

To configure your host for NVMe-oF you need to install the host and NVMe software packages, enable multipathing, and verify your host NQN configuration.

Steps

1. Install Proxmox VE 9.x on the server. After the installation is complete, verify that you are running the required Proxmox VE 9.x kernel:

   ```
   uname -r
   ```

   Example Proxmox VE 9.x kernel version:

   ```
   6.14.8-2-pve
   ```

2. Install the `nvme-cli` package:

   ```
   apt list|grep nvme-cli
   ```

   The following example shows an `nvme-cli` package version:

   ```
   nvme-cli/stable,now 2.13-2 amd64
   ```

3. Install the `libnvme` package:

   ```
   apt list|grep libnvme
   ```

   The following example shows an `libnvme` package version:

   ```
   libnvme-dev/stable 1.13-2 amd64
   ```

4. On the host, check the hostnqn string at `/etc/nvme/hostnqn`:

   ```
   cat /etc/nvme/hostnqn
   ```

The following example shows a `hostnqn` value:

```
nqn.2014-08.org.nvmexpress:uuid:39333550-3333-4753-4844-32594d4a524c
```

5. On the ONTAP system, verify that the `hostnqn` string matches the `hostnqn` string for the corresponding subsystem on the ONTAP array:

```
::> vserver nvme subsystem host show -vserver vs_proxmox_FC_NVMeFC
```

Show example

```
Vserver Subsystem Priority  Host NQN
------- --------- --------
-----------------------------------------------
vs_proxmox_FC_NVMeFC
        sub_176
                  regular   nqn.2014-
08.org.nvmexpress:uuid:39333550-3333-4753-4844-32594d4a4834
                  regular   nqn.2014-
08.org.nvmexpress:uuid:39333550-3333-4753-4844-32594d4a524c
2 entries were displayed
```

If the `hostnqn` strings do not match, use the `vserver modify` command to update the `hostnqn` string on your corresponding ONTAP storage system subsystem to match the `hostnqn` string from `/etc/nvme/hostnqn` on the host.

Step 2: Configure NVMe/FC and NVMe/TCP

Configure NVMe/FC with Broadcom/Emulex or Marvell/QLogic adapters, or configure NVMe/TCP using manual discovery and connect operations.

NVMe/FC - Broadcom/Emulex

Configure NVMe/FC for a Broadcom/Emulex adapter.

Steps

1. Verify that you are using the supported adapter model:
   a. Display the model names:

```
cat /sys/class/scsi_host/host*/modelname
```

   You should see the following output:

```
SN1700E2P
SN1700E2P
```

   b. Show the model descriptions:

```
cat /sys/class/scsi_host/host*/modeldesc
```

   You should see an output similar to the following example:

```
HPE SN1700E 64Gb 2p FC HBA
HPE SN1700E 64Gb 2p FC HBA
```

2. Verify that you are using the recommended Broadcom `lpfc` firmware and inbox driver:
   a. Display the firmware version:

```
cat /sys/class/scsi_host/host*/fwrev
```

   The command returns the firmware versions:

```
14.4.473.14, sli-4:6:d
14.4.473.14, sli-4:6:d
```

   b. Display the inbox driver version:

```
cat /sys/module/lpfc/version
```

   The following example shows a driver version:

```
0:14.4.0.7
```

For the current list of supported adapter driver and firmware versions, see the Interoperability Matrix Tool.

3. Verify that `lpfc_enable_fc4_type` is set to `3`:

```
cat /sys/module/lpfc/parameters/lpfc_enable_fc4_type
```

4. Verify that you can view your initiator ports:

```
cat /sys/class/fc_host/host*/port_name
```

You should see an output similar to:

```
0x10005ced8c531948
0x10005ced8c531949
```

5. Verify that your initiator ports are online:

```
cat /sys/class/fc_host/host*/port_state
```

You should see the following output:

```
Online
Online
```

6. Verify that the NVMe/FC initiator ports are enabled and that the target ports are visible:

```
cat /sys/class/scsi_host/host*/nvme_info
```

Show example

```
NVME Initiator Enabled
XRI Dist lpfc0 Total 6144 IO 5894 ELS 250
NVME LPORT lpfc0 WWPN x10005ced8c531948 WWNN x20005ced8c531948
DID x082400 ONLINE
NVME RPORT       WWPN x200ed039eac79573 WWNN x200dd039eac79573
DID x060902 TARGET DISCSRVC ONLINE
NVME RPORT       WWPN x2001d039eac79573 WWNN x2000d039eac79573
DID x060904 TARGET DISCSRVC ONLINE

NVME Statistics
LS: Xmt 0000000034 Cmpl 0000000034 Abort 00000000
LS XMIT: Err 00000000  CMPL: xb 00000000 Err 00000000
Total FCP Cmpl 0000000000142cfb Issue 0000000000142cfc OutIO
0000000000000001
        abort 00000005 noxri 00000000 nondlp 00000000 qdepth
00000000 wqerr 00000000 err 00000000
FCP CMPL: xb 00000005 Err 00000005

NVME Initiator Enabled
XRI Dist lpfc1 Total 6144 IO 5894 ELS 250
NVME LPORT lpfc1 WWPN x10005ced8c531949 WWNN x20005ced8c531949
DID x082500 ONLINE
NVME RPORT       WWPN x2010d039eac79573 WWNN x200dd039eac79573
DID x062902 TARGET DISCSRVC ONLINE
NVME RPORT       WWPN x2007d039eac79573 WWNN x2000d039eac79573
DID x062904 TARGET DISCSRVC ONLINE

NVME Statistics
LS: Xmt 0000000034 Cmpl 0000000034 Abort 00000000
LS XMIT: Err 00000000  CMPL: xb 00000000 Err 00000000
Total FCP Cmpl 00000000000d39f1 Issue 00000000000d39f2 OutIO
0000000000000001
        abort 00000005 noxri 00000000 nondlp 00000000 qdepth
00000000 wqerr 00000000 err 00000000
FCP CMPL: xb 00000005 Err 00000005
```

NVMe/FC - Marvell/QLogic

Configure NVMe/FC for a Marvell/QLogic adapter.

Steps

1. Verify that you are using the supported adapter driver and firmware versions:

```
cat /sys/class/fc_host/host*/symbolic_name
```

The following example shows driver and firmware versions:

```
SN1700Q FW:v9.15.05 DVR:v10.02.09.400-k
SN1700Q FW:v9.15.05 DVR:v10.02.09.400-k
```

2. Verify that `ql2xnvmeenable` is set. This enables the Marvell adapter to function as an NVMe/FC initiator:

```
cat /sys/module/qla2xxx/parameters/ql2xnvmeenable
```

The expected output is 1.

NVMe/TCP

The NVMe/TCP protocol doesn't support the auto-connect operation. Instead, you can discover the NVMe/TCP subsystems and namespaces by performing the NVMe/TCP `connect` or `connect-all` operations manually.

Steps

1. Check that the initiator port can get the discovery log page data across the supported NVMe/TCP LIFs:

```
nvme discover -t tcp -w host-traddr -a traddr
```

Show example

```
nvme discover -t tcp -w 192.168.165.72 -a 192.168.165.51
Discovery Log Number of Records 4, Generation counter 47
=====Discovery Log Entry 0======
trtype:  tcp
adrfam:  ipv4
subtype: current discovery subsystem
treq:    not specified
portid:  3
trsvcid: 8009
subnqn:  nqn.1992-
08.com.netapp:sn.c770be5d934811f0b624d039eac809ba:discovery
traddr:  192.168.165.51
eflags:  explicit discovery connections, duplicate discovery
information
sectype: none
=====Discovery Log Entry 1======
trtype:  tcp
adrfam:  ipv4
subtype: current discovery subsystem
treq:    not specified
portid:  1
trsvcid: 8009
subnqn:  nqn.1992-
08.com.netapp:sn.c770be5d934811f0b624d039eac809ba:discovery
traddr:  192.168.166.50
eflags:  explicit discovery connections, duplicate discovery
information
sectype: none
=====Discovery Log Entry 2======
trtype:  tcp
adrfam:  ipv4
subtype: nvme subsystem
treq:    not specified
portid:  3
trsvcid: 4420
subnqn:  nqn.1992-
08.com.netapp:sn.c770be5d934811f0b624d039eac809ba:subsystem.sub_
176
traddr:  192.168.165.51
eflags:  none
sectype: none
=====Discovery Log Entry 3======
trtype:  tcp
adrfam:  ipv4
```

```
subtype: nvme subsystem
treq:    not specified
portid:  1
trsvcid: 4420
subnqn:  nqn.1992-
08.com.netapp:sn.c770be5d934811f0b624d039eac809ba:subsystem.sub_
176
traddr:  192.168.166.50
eflags:  none
sectype: none
```

2. Run the `nvme connect-all` command across all the supported NVMe/TCP initiator-target LIFs across the nodes:

```
nvme connect-all -t tcp -w host-traddr -a traddr
```

Show example

```
nvme connect-all -t tcp -w 192.168.166.73 -a 192.168.166.50
nvme connect-all -t tcp -w 192.168.166.73 -a 192.168.166.51
nvme connect-all -t tcp -w 192.168.165.73 -a 192.168.165.50
nvme connect-all -t tcp -w 192.168.165.73 -a 192.168.165.51
```

The setting for the NVMe/TCP `ctrl_loss_tmo timeout` is automatically set to "off". As a result:

- There are no limits on the number of retries (indefinite retry).
- You don't need to manually configure a specific `ctrl_loss_tmo timeout` duration when using the `nvme connect` or `nvme connect-all` commands (option -l).
- The NVMe/TCP controllers don't experience timeouts in the event of a path failure and remain connected indefinitely.

Step 3: Optionally, enable 1MB I/O for NVMe/FC

ONTAP reports a Max Data Transfer Size (MDTS) of 8 in the Identify Controller data. This means the maximum I/O request size can be up to 1MB. To issue I/O requests of size 1MB for a Broadcom NVMe/FC host, you should increase the `lpfc` value of the `lpfc_sg_seg_cnt` parameter to 256 from the default value of 64.

These steps don't apply to Qlogic NVMe/FC hosts.

Steps

1. Set the `lpfc_sg_seg_cnt` parameter to 256:

```
cat /etc/modprobe.d/lpfc.conf
```

You should see an output similar to the following example:

```
options lpfc lpfc_sg_seg_cnt=256
```

2. Run the `update-initramfs` command, and reboot the host.
3. Verify that the value for `lpfc_sg_seg_cnt` is 256:

```
cat /sys/module/lpfc/parameters/lpfc_sg_seg_cnt
```

Step 4: Verify NVMe boot services

The `nvmefc-boot-connections.service` and `nvmf-autoconnect.service` boot services included in the NVMe/FC `nvme-cli` package are automatically enabled when the system boots.

After booting completes, verify that the `nvmefc-boot-connections.service` and `nvmf-autoconnect.service` boot services are enabled.

Steps

1. Verify that `nvmf-autoconnect.service` is enabled:

```
systemctl status nvmf-autoconnect.service
```

Show example output

```
○ nvmf-autoconnect.service - Connect NVMe-oF subsystems
automatically during boot
     Loaded: loaded (/usr/lib/systemd/system/nvmf-
autoconnect.service; enabled; preset: enabled)
     Active: inactive (dead) since Thu 2025-10-16 18:00:39 IST; 6
days ago
 Invocation: e146e0b2c339432aad6e0555a528872c
    Process: 1787 ExecStart=/usr/sbin/nvme connect-all
--context=autoconnect (code=exited, status=0/SUCCESS)
   Main PID: 1787 (code=exited, status=0/SUCCESS)
   Mem peak: 2.4M
        CPU: 12ms
Oct 16 18:00:39 HPE-DL365-14-176 systemd[1]: Starting nvmf-
autoconnect.service - Connect NVMe-oF subsystems automatically
during boot...
Oct 16 18:00:39 HPE-DL365-14-176 systemd[1]: nvmf-
autoconnect.service: Deactivated successfully.
Oct 16 18:00:39 HPE-DL365-14-176 systemd[1]: Finished nvmf-
autoconnect.service - Connect NVMe-oF subsystems automatically
during boot.
```

2. Verify that `nvmefc-boot-connections.service` is enabled:

```
systemctl status nvmefc-boot-connections.service
```

Show example output

```
nvmefc-boot-connections.service - Auto-connect to subsystems on FC-
NVME devices found during boot
     Loaded: loaded (/usr/lib/systemd/system/nvmefc-boot-
connections.service; enabled; preset: enabled)
     Active: inactive (dead) since Thu 2025-10-16 18:00:35 IST; 6
days ago
 Invocation: acf73ac1ef7a402198d6ecc4d075fab0
    Process: 1173 ExecStart=/bin/sh -c echo add >
/sys/class/fc/fc_udev_device/nvme_discovery (code=exited,
status=0/SUCCESS)
   Main PID: 1173 (code=exited, status=0/SUCCESS)
   Mem peak: 2.1M
        CPU: 11ms

Oct 16 18:00:35 HPE-DL365-14-176 systemd[1]: nvmefc-boot-
connections.service: Deactivated successfully.
Oct 16 18:00:35 HPE-DL365-14-176 systemd[1]: Finished nvmefc-boot-
connections.service - Auto-connect to subsystems on FC-NVME devices
found during boot.
```

Step 5: Verify the multipathing configuration

Verify that the in-kernel NVMe multipath status, ANA status, and ONTAP namespaces are correct for the NVMe-oF configuration.

Steps

1. Verify that the in-kernel NVMe multipath is enabled:

   ```
   cat /sys/module/nvme_core/parameters/multipath
   ```

   You should see the following output:

   ```
   Y
   ```

2. Verify that the appropriate NVMe-oF settings (such as, model set to NetApp ONTAP Controller and load balancing iopolicy set to round-robin) for the ONTAP namespaces show up correctly on the host:

   a. Show the subsystems:

      ```
      cat /sys/class/nvme-subsystem/nvme-subsys*/model
      ```

You should see the following output:

```
NetApp ONTAP Controller
NetApp ONTAP Controller
```

b. Display the policy:

```
cat /sys/class/nvme-subsystem/nvme-subsys*/iopolicy
```

You should see the following output:

```
queue-depth
queue-depth
```

3. Verify that the namespaces are created and correctly discovered on the host:

```
nvme list
```

Show example

```
Node          Generic     SN                   Model
Namespace  Usage                      Format           FW Rev
------------- ----------- --------------------
------------------------ ---------- --------------------------
---------------- --------
/dev/nvme2n1  /dev/ng2n1  81PqYFYq2aVAAAAAAAAB NetApp ONTAP
Controller   0x1         17.88  GB / 171.80  GB      4 KiB +  0 B
9.17.1
```

4. Verify that the controller state of each path is live and has the correct ANA status:

NVMe/FC

```
nvme list-subsys /dev/nvme3n9
```

Show example

```
nvme-subsys3 - NQN=nqn.1992-
08.com.netapp:sn.94929fdb84eb11f0b624d039eac809ba:subsystem.sub_
176
               hostnqn=nqn.2014-08.org.nvmexpress:uuid:39333550-
3333-4753-4844-32594d4a524c
\
 +- nvme1 fc traddr=nn-0x200dd039eac79573:pn-
0x2010d039eac79573,host_traddr=nn-0x20005ced8c531949:pn-
0x10005ced8c531949 live optimized
 +- nvme3 fc traddr=nn-0x200dd039eac79573:pn-
0x200ed039eac79573,host_traddr=nn-0x20005ced8c531948:pn-
0x10005ced8c531948 live optimized
+- nvme5 fc traddr=nn-0x200dd039eac79573:pn-
0x200fd039eac79573,host_traddr=nn-0x20005ced8c531949:pn-
0x10005ced8c531949 live non-optimized
 +- nvme7 fc traddr=nn-0x200dd039eac79573:pn-
0x2011d039eac79573,host_traddr=nn-0x20005ced8c531948:pn-
0x10005ced8c531948 live non-optimized
```

NVMe/TCP

```
nvme list-subsys /dev/nvme2n3
```

Show example

```
nvme-subsys2 - NQN=nqn.1992-
08.com.netapp:sn.c770be5d934811f0b624d039eac809ba:subsystem.sub_
176
               hostnqn=nqn.2014-08.org.nvmexpress:uuid:39333550-
3333-4753-4844-32594d4a524c
\
 +- nvme2 tcp
traddr=192.168.166.50,trsvcid=4420,host_traddr=192.168.166.73,sr
c_addr=192.168.166.73 live optimized
 +- nvme4 tcp
traddr=192.168.165.51,trsvcid=4420,host_traddr=192.168.165.73,sr
c_addr=192.168.165.73 live optimized
+- nvme6 tcp
traddr=192.168.166.51,trsvcid=4420,host_traddr=192.168.166.73,sr
c_addr=192.168.166.73 live non-optimized
 +- nvme8 tcp
traddr=192.168.165.50,trsvcid=4420,host_traddr=192.168.165.73,sr
c_addr=192.168.165.73 live non-optimized
```

5. Verify that the NetApp plug-in displays the correct values for each ONTAP namespace device:

Column

```
nvme netapp ontapdevices -o column
```

Show example

```
Device        Vserver               Namespace Path
------------- ---------------------
------------------------------
/dev/nvme2n9  vs_proxmox_FC_NVMeFC  /vol/vol_180_data_nvmefc4/ns

NSID       UUID                                   Size
---- ------------------------------------  --------
1    e3d3d544-de8b-4787-93af-bfec7769e909  32.21GB
```

JSON

```
nvme netapp ontapdevices -o json
```

Show example

```
{
      "Device":"/dev/nvme2n9",
      "Vserver":"vs_proxmox_FC_NVMeFC",
      "Subsystem":"sub_176",
      "Namespace_Path":"/vol/vol_180_data_nvmefc4/ns",
      "NSID":9,
      "UUID":"e3d3d544-de8b-4787-93af-bfec7769e909",
      "LBA_Size":4096,
      "Namespace_Size":32212254720,
      "UsedBytes":67899392,
      "Version":"9.17.1"
    }
  ]
```

Step 6: Review the known issues

There are no known issues.

Configure Proxmox VE 8.x for NVMe-oF with ONTAP storage

The Proxmox VE 8.x host supports the NVMe over Fibre Channel (NVMe/FC) and NVMe over TCP (NVMe/TCP) protocols with Asymmetric Namespace Access (ANA). ANA provides multipathing functionality equivalent to asymmetric logical unit access (ALUA) in iSCSI and FCP environments.

Learn how to configure NVMe over Fabrics (NVMe-oF) hosts for Proxmox VE 8.x. For more support and feature information, see ONTAP support and features.

NVMe-oF with Proxmox VE 8.x has the following known limitation:

- The SAN boot configuration for NVMe-FC is not supported.

Step 1: Install Proxmox VE and NVMe software and verify your configuration

To configure your host for NVMe-oF you need to install the host and NVMe software packages, enable multipathing, and verify your host NQN configuration.

Steps

1. Install Proxmox 8.x on the server. After the installation is complete, verify that you are running the specified Proxmox 8.x kernel:

   ```
   uname -r
   ```

   The following example shows a Proxmox kernel version:

   ```
   6.8.12-10-pve
   ```

2. Install the `nvme-cli` package:

   ```
   apt list|grep nvme-cli
   ```

   The following example shows an `nvme-cli` package version:

   ```
   nvme-cli/oldstable,now 2.4+really2.3-3 amd64
   ```

3. Install the `libnvme` package:

   ```
   apt list|grep libnvme
   ```

   The following example shows an `libnvme` package version:

```
libnvme1/oldstable,now 1.3-1+deb12u1 amd64
```

4. On the host, check the hostnqn string at `/etc/nvme/hostnqn`:

```
cat /etc/nvme/hostnqn
```

The following example shows a `hostnqn` value:

```
nqn.2014-08.org.nvmexpress:uuid:1536c9a6-f954-11ea-b24d-0a94efb46eaf
```

5. On the ONTAP system, verify that the `hostnqn` string matches the `hostnqn` string for the corresponding subsystem on the ONTAP array:

```
::>  vserver nvme subsystem host show -vserver proxmox_120_122
```

Show example

```
Vserver Subsystem Priority  Host NQN
------- --------- --------  ---------
proxmox_120_122
proxmox_120_122
                  regular   nqn.2014-
08.org.nvmexpress:uuid:1536c9a6-f954-11ea-b24d-0a94efb46eaf
                  regular   nqn.2014-
08.org.nvmexpress:uuid:991a7476-f9bf-11ea-8b73-0a94efb46c3b
proxmox_120_122_tcp
                  regular   nqn.2014-
08.org.nvmexpress:uuid:1536c9a6-f954-11ea-b24d-0a94efb46eaf
                  regular  nqn.2014-08.org.nvmexpress:uuid:991a7476-
f9bf-11ea-8b73-0a94efb46c3b
2 entries were displayed.
```

If the `hostnqn` strings do not match, use the `vserver modify` command to update the `hostnqn` string on your corresponding ONTAP storage system subsystem to match the `hostnqn` string from `/etc/nvme/hostnqn` on the host.

Step 2: Configure NVMe/FC and NVMe/TCP

Configure NVMe/FC with Broadcom/Emulex or Marvell/QLogic adapters, or configure NVMe/TCP using manual discovery and connect operations.

NVMe/FC - Broadcom/Emulex

Configure NVMe/FC for a Broadcom/Emulex adapter.

Steps

1. Verify that you are using the supported adapter model:
   a. Display the model names:

```
cat /sys/class/scsi_host/host*/modelname
```

   You should see the following output:

```
LPe35002-M2
LPe35002-M2
```

   b. Display the model descriptions:

```
cat /sys/class/scsi_host/host*/modeldesc
```

   You should see an output similar to the following example:

```
Emulex LPe35002-M2 2-Port 32Gb Fibre Channel Adapter
Emulex LPe35002-M2 2-Port 32Gb Fibre Channel Adapter
```

2. Verify that you are using the recommended Broadcom `lpfc` firmware and inbox driver:
   a. Display the firmware version:

```
cat /sys/class/scsi_host/host*/fwrev
```

   The command returns the firmware versions:

```
14.0.505.12, sli-4:6:d
14.0.505.12, sli-4:6:d
```

   b. Display the inbox driver version:

```
cat /sys/module/lpfc/version
```

   The following example shows a driver version:

```
0:14.2.0.17
```

For the current list of supported adapter driver and firmware versions, see the Interoperability Matrix Tool.

3. Verify that `lpfc_enable_fc4_type` is set to `3`:

```
cat /sys/module/lpfc/parameters/lpfc_enable_fc4_type
```

4. Verify that you can view your initiator ports:

```
cat /sys/class/fc_host/host*/port_name
```

You should see an output similar to:

```
0x100000109b95467e
0x100000109b95467f
```

5. Verify that your initiator ports are online:

```
cat /sys/class/fc_host/host*/port_state
```

You should see the following output:

```
Online
Online
```

6. Verify that the NVMe/FC initiator ports are enabled and that the target ports are visible:

```
cat /sys/class/scsi_host/host*/nvme_info
```

Show example

```
NVME Initiator Enabled
XRI Dist lpfc0 Total 6144 IO 5894 ELS 250
NVME LPORT lpfc0 WWPN x10005ced8c531948 WWNN x20005ced8c531948
DID x082400
ONLINE
NVME RPORT WWPN x200ed039eac79573 WWNN x200dd039eac79573 DID
x060902
TARGET DISCSRVC ONLINE
NVME RPORT WWPN x2001d039eac79573 WWNN x2000d039eac79573 DID
x060904
TARGET DISCSRVC ONLINE
NVME Statistics LS: Xmt 0000000034 Cmpl 0000000034 Abort
00000000 LS XMIT: Err 00000000 CMPL: xb 00000000 Err 00000000
Total FCP Cmpl 0000000000142cfb Issue 0000000000142cfc OutIO
0000000000000001 abort 00000005 noxri 00000000 nondlp 00000000
qdepth 00000000 wqerr 00000000 err 00000000 FCP CMPL: xb
00000005 Err 00000005 NVME Initiator Enabled XRI Dist lpfc1
Total 6144 IO 5894 ELS 250 NVME LPORT lpfc1 WWPN
x10005ced8c531949 WWNN x20005ced8c531949 DID x082500
ONLINE
NVME RPORT WWPN x2010d039eac79573 WWNN x200dd039eac79573 DID
x062902
TARGET DISCSRVC ONLINE
NVME RPORT WWPN x2007d039eac79573 WWNN x2000d039eac79573 DID
x062904
TARGET DISCSRVC ONLINE
NVME Statistics LS: Xmt 0000000034 Cmpl 0000000034 Abort
00000000 LS XMIT: Err 00000000 CMPL: xb 00000000 Err 00000000
Total FCP Cmpl 00000000000d39f1 Issue 00000000000d39f2 OutIO
0000000000000001 abort 00000005 noxri 00000000 nondlp 00000000
qdepth 00000000 wqerr 00000000 err 00000000 FCP CMPL: xb
00000005 Err 00000005
```

NVMe/FC - Marvell/QLogic

Configure NVMe/FC for a Marvell/QLogic adapter.

1. Verify that you are using the supported adapter driver and firmware versions:

```
cat /sys/class/fc_host/host*/symbolic_name
```

The following example shows driver and firmware versions:

```
QLE2872 FW:v9.15.00 DVR:v10.02.09.300-k
QLE2872 FW:v9.15.00 DVR:v10.02.09.300-k
```

2. Verify that `ql2xnvmeenable` is set. This enables the Marvell adapter to function as an NVMe/FC initiator:

```
cat /sys/module/qla2xxx/parameters/ql2xnvmeenable
```

The expected output is 1.

NVMe/TCP

The NVMe/TCP protocol doesn't support the auto-connect operation. Instead, you can discover the NVMe/TCP subsystems and namespaces by performing the NVMe/TCP `connect` or `connect-all` operations manually.

Steps

1. Check that the initiator port can get the discovery log page data across the supported NVMe/TCP LIFs:

```
nvme discover -t tcp -w host-traddr -a traddr
```

Show example

```
nvme discover -t tcp -w 192.168.2.22 -a 192.168.2.30

Discovery Log Number of Records 12, Generation counter 13
=====Discovery Log Entry 0======
trtype:  tcp
adrfam:  ipv4
subtype: current discovery subsystem
treq:    not specified
portid:  10
trsvcid: 8009
subnqn:  nqn.1992-
08.com.netapp:sn.ae9f2d55a7ec11ef8751d039ea9e891c:discovery
traddr:  192.168.2.30
eflags:  explicit discovery connections, duplicate discovery
information
sectype: none
=====Discovery Log Entry 1======
trtype:  tcp
adrfam:  ipv4
subtype: current discovery subsystem
treq:    not specified
portid:  9
trsvcid: 8009
subnqn:  nqn.1992-
08.com.netapp:sn.ae9f2d55a7ec11ef8751d039ea9e891c:discovery
traddr:  192.168.1.30
eflags:  explicit discovery connections, duplicate discovery
information
sectype: none
=====Discovery Log Entry 2======
trtype:  tcp
adrfam:  ipv4
subtype: current discovery subsystem
treq:    not specified
portid:  12
trsvcid: 8009
subnqn:  nqn.1992-
08.com.netapp:sn.ae9f2d55a7ec11ef8751d039ea9e891c:discovery
traddr:  192.168.2.25
eflags:  explicit discovery connections, duplicate discovery
information
sectype: none
=====Discovery Log Entry 3======
trtype:  tcp
```

```
adrfam:  ipv4
subtype: current discovery subsystem
treq:    not specified
portid:  11
trsvcid: 8009
subnqn:  nqn.1992-
08.com.netapp:sn.ae9f2d55a7ec11ef8751d039ea9e891c:discovery
traddr:  192.168.1.25
eflags:  explicit discovery connections, duplicate discovery
information
sectype: none
=====Discovery Log Entry 4======
trtype:  tcp
adrfam:  ipv4
subtype: nvme subsystem
treq:    not specified
portid:  10
trsvcid: 4420
subnqn:  nqn.1992-
08.com.netapp:sn.ae9f2d55a7ec11ef8751d039ea9e891c:subsystem.prox
mox_120_122
traddr:  192.168.2.30
eflags:  none
sectype: none
=====Discovery Log Entry 5======
trtype:  tcp
adrfam:  ipv4
subtype: nvme subsystem
treq:    not specified
portid:  9
trsvcid: 4420
subnqn:  nqn.1992-
08.com.netapp:sn.ae9f2d55a7ec11ef8751d039ea9e891c:subsystem.prox
mox_120_122
traddr:  192.168.1.30
eflags:  none
sectype: none
=====Discovery Log Entry 6======
trtype:  tcp
adrfam:  ipv4
subtype: nvme subsystem
treq:    not specified
portid:  12
trsvcid: 4420
subnqn:  nqn.1992-
08.com.netapp:sn.ae9f2d55a7ec11ef8751d039ea9e891c:subsystem.prox
```

```
mox_120_122
traddr:  192.168.2.25
eflags:  none
sectype: none
=====Discovery Log Entry 7======
trtype:  tcp
adrfam:  ipv4
subtype: nvme subsystem
treq:    not specified
portid:  11
trsvcid: 4420
subnqn:  nqn.1992-
08.com.netapp:sn.ae9f2d55a7ec11ef8751d039ea9e891c:subsystem.prox
mox_120_122
traddr:  192.168.1.25
eflags:  none
sectype: none
=====Discovery Log Entry 8======
trtype:  tcp
adrfam:  ipv4
subtype: nvme subsystem
treq:    not specified
portid:  10
trsvcid: 4420
subnqn:  nqn.1992-
08.com.netapp:sn.ae9f2d55a7ec11ef8751d039ea9e891c:subsystem.prox
mox_120_122_tcp
traddr:  192.168.2.30
eflags:  none
sectype: none
=====Discovery Log Entry 9======
trtype:  tcp
adrfam:  ipv4
subtype: nvme subsystem
treq:    not specified
portid:  9
trsvcid: 4420
subnqn:  nqn.1992-
08.com.netapp:sn.ae9f2d55a7ec11ef8751d039ea9e891c:subsystem.prox
mox_120_122_tcp
traddr:  192.168.1.30
eflags:  none
sectype: none
=====Discovery Log Entry 10======
trtype:  tcp
adrfam:  ipv4
```

```
subtype: nvme subsystem
treq:    not specified
portid:  12
trsvcid: 4420
subnqn:  nqn.1992-
08.com.netapp:sn.ae9f2d55a7ec11ef8751d039ea9e891c:subsystem.prox
mox_120_122_tcp
traddr:  192.168.2.25
eflags:  none
sectype: none
=====Discovery Log Entry 11======
trtype:  tcp
adrfam:  ipv4
subtype: nvme subsystem
treq:    not specified
portid:  11
trsvcid: 4420
subnqn:  nqn.1992-
08.com.netapp:sn.ae9f2d55a7ec11ef8751d039ea9e891c:subsystem.prox
mox_120_122_tcp
traddr:  192.168.1.25
eflags:  none
sectype: none
```

2. Verify that the other NVMe/TCP initiator-target LIF combinations can successfully retrieve discovery log page data:

```
nvme discover -t tcp -w host-traddr -a traddr
```

Show example

```
nvme discover -t tcp -w 192.168.1.22 -a 192.168.1.30
nvme discover -t tcp -w 192.168.2.22 -a 192.168.2.30
nvme discover -t tcp -w 192.168.1.22 -a 192.168.1.25
nvme discover -t tcp -w 192.168.2.22 -a 192.168.2.25
```

3. Run the `nvme connect-all` command across all the supported NVMe/TCP initiator-target LIFs across the nodes:

```
nvme connect-all -t tcp -w host-traddr -a traddr
```

Show example

```
nvme connect-all -t tcp -w 192.168.1.22 -a 192.168.1.30
nvme connect-all -t tcp -w 192.168.2.22 -a 192.168.2.30
nvme connect-all -t tcp -w 192.168.1.22 -a 192.168.1.25
nvme connect-all -t tcp -w 192.168.2.22 -a 192.168.2.25
```

The setting for the NVMe/TCP `ctrl_loss_tmo timeout` is automatically set to "off". As a result:

- There are no limits on the number of retries (indefinite retry).
- You don't need to manually configure a specific `ctrl_loss_tmo timeout` duration when using the `nvme connect` or `nvme connect-all` commands (option -l).
- The NVMe/TCP controllers don't experience timeouts in the event of a path failure and remain connected indefinitely.

Step 3: Optionally, enable 1MB I/O for NVMe/FC

ONTAP reports a Max Data Transfer Size (MDTS) of 8 in the Identify Controller data. This means the maximum I/O request size can be up to 1MB. To issue I/O requests of size 1MB for a Broadcom NVMe/FC host, you should increase the `lpfc` value of the `lpfc_sg_seg_cnt` parameter to 256 from the default value of 64.

These steps don't apply to Qlogic NVMe/FC hosts.

Steps

1. Set the `lpfc_sg_seg_cnt` parameter to 256:

   ```
   cat /etc/modprobe.d/lpfc.conf
   ```

   You should see an output similar to the following example:

   ```
   options lpfc lpfc_sg_seg_cnt=256
   ```

2. Run the `update-initramfs` command, and reboot the host.
3. Verify that the value for `lpfc_sg_seg_cnt` is 256:

   ```
   cat /sys/module/lpfc/parameters/lpfc_sg_seg_cnt
   ```

Step 4: Verify NVMe boot services

With Proxmox 8.x, the `nvmefc-boot-connections.service` and `nvmf-autoconnect.service` boot services included in the NVMe/FC `nvme-cli` package are automatically enabled when the system boots.

After booting completes, verify that the `nvmefc-boot-connections.service` and `nvmf-autoconnect.service` boot services are enabled.

Steps

1. Verify that `nvmf-autoconnect.service` is enabled:

```
systemctl status nvmf-autoconnect.service
```

Show example output

```
○ nvmf-autoconnect.service - Connect NVMe-oF subsystems
automatically during boot
     Loaded: loaded (/lib/systemd/system/nvmf-autoconnect.service;
enabled; preset: enabled)
     Active: inactive (dead) since Fri 2025-11-21 19:59:10 IST; 8s
ago
    Process: 256613 ExecStartPre=/sbin/modprobe nvme-fabrics
(code=exited, status=0/SUCCESS)
    Process: 256614 ExecStart=/usr/sbin/nvme connect-all
(code=exited, status=0/SUCCESS)
   Main PID: 256614 (code=exited, status=0/SUCCESS)
        CPU: 18ms
Nov 21 19:59:07 SR665-14-122.lab.eng.btc.netapp.in systemd[1]:
Starting nvmf-autoconnect.service - Connect NVMe-oF subsystems
automatically during boot...
Nov 21 19:59:10 SR665-14-122.lab.eng.btc.netapp.in nvme[256614]:
Failed to write to /dev/nvme-fabrics: Invalid argument
Nov 21 19:59:10 SR665-14-122.lab.eng.btc.netapp.in nvme[256614]:
Failed to write to /dev/nvme-fabrics: Invalid argument
Nov 21 19:59:10 SR665-14-122.lab.eng.btc.netapp.in systemd[1]: nvmf-
autoconnect.service: Deactivated successfully.
Nov 21 19:59:10 SR665-14-122.lab.eng.btc.netapp.in systemd[1]:
Finished nvmf-autoconnect.service - Connect NVMe-oF subsystems
automatically during boot.
```

2. Verify that `nvmefc-boot-connections.service` is enabled:

```
systemctl status nvmefc-boot-connections.service
```

Show example output

```
○ nvmefc-boot-connections.service - Auto-connect to subsystems on
FC-NVME devices found during boot
    Loaded: loaded (/lib/systemd/system/nvmefc-boot-
connections.service; enabled; preset: enabled)
     Active: inactive (dead) since Thu 2025-11-20 17:48:29 IST; 1
day 2h ago
    Process: 1381 ExecStart=/bin/sh -c echo add >
/sys/class/fc/fc_udev_device/nvme_discovery (code=exited,
status=0/SUCCESS)
   Main PID: 1381 (code=exited, status=0/SUCCESS)
        CPU: 3ms

Nov 20 17:48:29 SR665-14-122.lab.eng.btc.netapp.in systemd[1]:
Starting nvmefc-boot-connections.service - Auto-connect to
subsystems on FC-NVME devices found during boot..
Nov 20 17:48:29 SR665-14-122.lab.eng.btc.netapp.in systemd[1]:
nvmefc-boot-connections.service: Deactivated successfully.
Nov 20 17:48:29 SR665-14-122.lab.eng.btc.netapp.in systemd[1]:
Finished nvmefc-boot-connections.service - Auto-connect to
subsystems on FC-NVME devices found during boot...
```

Step 5: Verify the multipathing configuration

Verify that the in-kernel NVMe multipath status, ANA status, and ONTAP namespaces are correct for the NVMe-oF configuration.

Steps

1. Verify that the in-kernel NVMe multipath is enabled:

   ```
   cat /sys/module/nvme_core/parameters/multipath
   ```

   You should see the following output:

   ```
   Y
   ```

2. Verify that the appropriate NVMe-oF settings (such as, model set to NetApp ONTAP Controller and load balancing iopolicy set to round-robin) for the ONTAP namespaces show up correctly on the host:

   a. Show the subsystems:

      ```
      cat /sys/class/nvme-subsystem/nvme-subsys*/model
      ```

You should see the following output:

```
NetApp ONTAP Controller
NetApp ONTAP Controller
```

b. Display the policy:

```
cat /sys/class/nvme-subsystem/nvme-subsys*/iopolicy
```

You should see the following output:

```
round-robin
round-robin
```

3. Verify that the namespaces are created and correctly discovered on the host:

```
nvme list
```

Show example

```
Node                  Generic               SN
Model                                    Namespace  Usage
Format           FW Rev
--------------------- --------------------- --------------------
---------------------------------------- ----------
-------------------------- ---------------- --------
/dev/nvme2n20         /dev/ng2n20           81K13BUDdygsAAAAAAAG
NetApp ONTAP Controller                  10          5.56  GB /
91.27  GB      4 KiB +  0 B   9.18.1
```

4. Verify that the controller state of each path is live and has the correct ANA status:

NVMe/FC

```
nvme list-subsys /dev/nvme2n20
```

Show example

```
nvme-subsys2 - NQN= nqn.1992-
08.com.netapp:sn.ae9f2d55a7ec11ef8751d039ea9e891c:subsystem.prox
mox_120_122_tcp
\
+- nvme1 fc traddr=nn-0x200dd039eac79573:pn-
0x2010d039eac79573,host_traddr=nn-0x20005ced8c531949:pn-
0x10005ced8c531949 live optimized
+- nvme3 fc traddr=nn-0x200dd039eac79573:pn-
0x200ed039eac79573,host_traddr=nn-0x20005ced8c531948:pn-
0x10005ced8c531948 live optimized
+- nvme5 fc traddr=nn-0x200dd039eac79573:pn-
0x200fd039eac79573,host_traddr=nn-0x20005ced8c531949:pn-
0x10005ced8c531949 live non-optimized
+- nvme7 fc traddr=nn-0x200dd039eac79573:pn-
0x2011d039eac79573,host_traddr=nn-0x20005ced8c531948:pn-
0x10005ced8c531948 live non-optimized
```

NVMe/TCP

```
nvme list-subsys /dev/nvme2n3
```

Show example

```
nvme-subsys2 - NQN= qn.1992-
08.com.netapp:sn.ae9f2d55a7ec11ef8751d039ea9e891c:subsystem.prox
mox_120_122_tcp
\
+- nvme2 tcp
traddr=192.168.1.30,trsvcid=4420,host_traddr=192.168.1.22,src_ad
dr=192.168.1.22 live optimized
+- nvme4 tcp
traddr=192.168.2.30,trsvcid=4420,host_traddr=192.168.2.22,src_ad
dr=192.168.2.22 live optimized
+- nvme6 tcp
traddr=192.168.1.25,trsvcid=4420,host_traddr=192.168.1.22,src_ad
dr=192.168.1.22 live non-optimized
+- nvme8 tcp
traddr=192.168.2.25,trsvcid=4420,host_traddr=192.168.2.22,src_ad
dr=192.168.2.22 live non-optimized
```

5. Verify that the NetApp plug-in displays the correct values for each ONTAP namespace device:

Column

```
nvme netapp ontapdevices -o column
```

Show example

```
Device        Vserver               Namespace Path
------------- ---------------------
------------------------------
/dev/nvme2n11     proxmox_120_122 / /vol/vm120_tcp1/ns

NSID       UUID                                  Size
---- ------------------------------------  --------
1          5aefea74-f0cf-4794-a7e9-e113c4659aca   37.58GB
```

JSON

```
nvme netapp ontapdevices -o json
```

Show example

```
{
      "Device":"/dev/nvme2n11",
      "Vserver":"proxmox_120_122",
      "Namespace_Path":"/vol/vm120_tcp1/ns",
      "NSID":1,
      "UUID":"5aefea74-f0cf-4794-a7e9-e113c4659aca",
       “Size”:”37.58GB”,
      "LBA_Data_Size":4096,
      "Namespace_Size":32212254720
    }
  ]
```

Step 6: Review the known issues

There are no known issues.

RHEL

Learn about ONTAP support and features for RHEL hosts

Features supported for host configuration with NVMe over Fabrics (NVMe-oF) varies based on your version of ONTAP and RHEL.

Feature	RHEL host version	ONTAP version
Secure in-band authentication is supported over NVMe/TCP between a RHEL host and an ONTAP controller	9.3 or later	9.12.1 or later
NVMe/TCP provides namespaces using the native `nvme-cli` package	8.2 or later	9.10.1 or later
NVMe/TCP is a fully supported enterprise feature	9.0 or later	9.10.1 or later
NVMe and SCSI traffic is supported on the same host using NVMe multipath for NVMe-oF namespaces and dm-multipath for SCSI LUNs	8.2 or later	9.4 or later

ONTAP supports the following SAN host features regardless of the ONTAP version running on your system setup.

Feature	RHEL host version
Native NVMe multipathing is enabled by default	10.0 or later
The native udev rule in the `nvme-cli` package provides queue-depth load balancing for NVMe multipathing	9.6 or later
SAN booting is enabled using the NVMe/FC protocol	9.4 or later
The `nvme-cli` package includes auto-connect scripts removing the need for third-party scripts	8.2 or later
The native udev rule in the `nvme-cli` package provides round-robin load balancing for NVMe multipathing	8.2 or later

For details on supported configurations, see the Interoperability Matrix Tool.

What's next

If your RHEL version is …	Learn about …
10 series	Configuring NVMe for RHEL 10.x
9 series	Configuring NVMe for RHEL 9.x
8 series	Configuring NVMe for RHEL 8.x

Related information

Learn about managing NVMe protocols

Configure RHEL 10.x for NVMe-oF with ONTAP storage

Red Hat Enterpirse Linux (RHEL) hosts support the NVMe over Fibre Channel (NVMe/FC) and NVMe over TCP (NVMe/TCP) protocols with Asymmetric Namespace Access (ANA). ANA provides multipathing functionality equivalent to asymmetric logical unit access (ALUA) in iSCSI and FCP environments.

Learn how to configure NVMe over Fabrics (NVMe-oF) hosts for RHEL 10.x. For more support and feature information, see RHEL ONTAP support and features.

NVMe-oF with RHEL 10.x has the following known limitations:

- The `nvme disconnect-all` command disconnects both root and data filesystems and might lead to system instability. Do not issue this on systems booting from SAN over NVMe-TCP or NVMe-FC namespaces.

Step 1: Optionally, enable SAN booting

You can configure your host to use SAN booting to simplify deployment and improve scalability. Use the Interoperability Matrix Tool to verify that your Linux OS, host bus adapter (HBA), HBA firmware, HBA boot BIOS, and ONTAP version support SAN booting.

Steps

1. Create a NVMe namespace and map it to the host.
2. Enable SAN booting in the server BIOS for the ports to which the SAN boot namespace is mapped.

   For information on how to enable the HBA BIOS, see your vendor-specific documentation.

3. Reboot the host and verify that the OS is up and running.

Step 2: Install RHEL and NVMe software and verify your configuration

To configure your host for NVMe-oF you need to install the host and NVMe software packages, enable multipathing, and verify your host NQN configuration.

Steps

1. Install RHEL 10.x on the server. After the installation is complete, verify that you are running the required RHEL 10.x kernel:

   ```
   uname -r
   ```

   Example RHEL kernel version:

   ```
   6.12.0-124.8.1.el10_1.x86_64
   ```

2. Install the `nvme-cli` package:

```
rpm -qa|grep nvme-cli
```

The following example shows an `nvme-cli` package version:

```
nvme-cli-2.13-2.el10.x86_64
```

3. Install the `libnvme` package:

```
rpm -qa|grep libnvme
```

The following example shows an `libnvme` package version:

```
libnvme-1.13-1.el10.x86_64
```

4. On the host, check the hostnqn string at `/etc/nvme/hostnqn`:

```
cat /etc/nvme/hostnqn
```

The following example shows an `hostnqn` version:

```
nqn.2014-08.org.nvmexpress:uuid:4c4c4544-0056-5410-8048-c7c04f425633
```

5. On the ONTAP system, verify that the `hostnqn` string matches the `hostnqn` string for the corresponding subsystem on the ONTAP storage system:

```
::> vserver nvme subsystem host show -vserver vs_coexistence_QLE2872
```

Show example

```
Vserver Subsystem Priority  Host NQN
------- --------- --------
-------------------------------------------------
vs_coexistence_QLE2872
               subsystem_1
                         regular   nqn.2014-
08.org.nvmexpress:uuid:4c4c4544-0056-5410-8048-c7c04f425633
               subsystem_10
                         regular   nqn.2014-
08.org.nvmexpress:uuid:4c4c4544-0056-5410-8048-c7c04f425633
               subsystem_11
                         regular   nqn.2014-
08.org.nvmexpress:uuid:4c4c4544-0056-5410-8048-c7c04f425633
```

If the `hostnqn` strings do not match, use the `vserver modify` command to update the `hostnqn` string on your corresponding ONTAP storage system subsystem to match the `hostnqn` string from `/etc/nvme/hostnqn` on the host.

Step 3: Configure NVMe/FC and NVMe/TCP

Configure NVMe/FC with Broadcom/Emulex or Marvell/QLogic adapters, or configure NVMe/TCP using manual discovery and connect operations.

NVMe/FC - Broadcom/Emulex

Configure NVMe/FC for a Broadcom/Emulex adapter.

Steps

1. Verify that you are using the supported adapter model:
   a. Display the model names:

```
cat /sys/class/scsi_host/host*/modelname
```

   You should see the following output:

```
SN1700E2P
SN1700E2P
```

   b. Display the model descriptions:

```
cat /sys/class/scsi_host/host*/modeldesc
```

   You should see an output similar to the following example:

```
HPE SN1700E 64Gb 2p FC HBA
HPE SN1700E 64Gb 2p FC HBA
```

2. Verify that you are using the recommended Broadcom `lpfc` firmware and inbox driver:
   a. Display the firmware version:

```
cat /sys/class/scsi_host/host*/fwrev
```

   The command returns the firmware versions:

```
14.4.393.25, sli-4:6:d
14.4.393.25, sli-4:6:d
```

   b. Display the inbox driver version:

```
cat /sys/module/lpfc/version
```

   The following example shows a driver version:

```
0:14.4.0.9
```

For the current list of supported adapter driver and firmware versions, see the Interoperability Matrix Tool.

3. Verify that `lpfc_enable_fc4_type` is set to `3`:

```
cat /sys/module/lpfc/parameters/lpfc_enable_fc4_type
```

4. Verify that you can view your initiator ports:

```
cat /sys/class/fc_host/host*/port_name
```

You should see an output similar to:

```
0x10005cba2cfca7de
0x10005cba2cfca7df
```

5. Verify that your initiator ports are online:

```
cat /sys/class/fc_host/host*/port_state
```

You should see the following output:

```
Online
Online
```

6. Verify that the NVMe/FC initiator ports are enabled and that the target ports are visible:

```
cat /sys/class/scsi_host/host*/nvme_info
```

Show example

```
NVME Initiator Enabled
XRI Dist lpfc0 Total 6144 IO 5894 ELS 250
NVME LPORT lpfc0 WWPN x10005cba2cfca7de WWNN x20005cba2cfca7de
DID x080f00 ONLINE
NVME RPORT       WWPN x2023d039eac03c33 WWNN x2021d039eac03c33
DID x082209 TARGET DISCSRVC ONLINE
NVME RPORT       WWPN x200ed039eac03c33 WWNN x200cd039eac03c33
DID x082203 TARGET DISCSRVC ONLINE
NVME RPORT       WWPN x2022d039eac03c33 WWNN x2021d039eac03c33
DID x082609 TARGET DISCSRVC ONLINE
NVME RPORT       WWPN x200dd039eac03c33 WWNN x200cd039eac03c33
DID x082604 TARGET DISCSRVC ONLINE

NVME Statistics
LS: Xmt 0000000501 Cmpl 0000000501 Abort 00000000
LS XMIT: Err 00000000  CMPL: xb 00000000 Err 00000000
Total FCP Cmpl 00000000000583b7 Issue 000000000005840d OutIO
0000000000000056
abort 0000010f noxri 00000000 nondlp 00000000 qdepth 00000000
wqerr 00000000   err 00000000
FCP CMPL: xb 0000010f Err 0000010f

NVME Initiator Enabled
XRI Dist lpfc1 Total 6144 IO 5894 ELS 250
NVME LPORT lpfc1 WWPN x10005cba2cfca7df WWNN x20005cba2cfca7df
DID x080b00 ONLINE
NVME RPORT       WWPN x2024d039eac03c33 WWNN x2021d039eac03c33
DID x082309 TARGET DISCSRVC ONLINE
NVME RPORT       WWPN x200fd039eac03c33 WWNN x200cd039eac03c33
DID x082304 TARGET DISCSRVC ONLINE
NVME RPORT       WWPN x2025d039eac03c33 WWNN x2021d039eac03c33
DID x082708 TARGET DISCSRVC ONLINE
NVME RPORT       WWPN x2010d039eac03c33 WWNN x200cd039eac03c33
DID x082703 TARGET DISCSRVC ONLINE

NVME Statistics
LS: Xmt 00000006eb Cmpl 00000006eb Abort 00000000
LS XMIT: Err 00000000  CMPL: xb 00000000 Err 00000000
Total FCP Cmpl 000000000004d600 Issue 000000000004d65f OutIO
000000000000005f
abort 000001c1 noxri 00000000 nondlp 00000000 qdepth 00000000
wqerr 00000000 err 00000000
FCP CMPL: xb 000001c1 Err 000001c2
```

NVMe/FC - Marvell/QLogic

Configure NVMe/FC for a Marvell/QLogic adapter.

Steps

1. Verify that you are using the supported adapter driver and firmware versions:

```
cat /sys/class/fc_host/host*/symbolic_name
```

   The following example shows driver and firmware versions:

```
QLE2872 FW:v9.15.06 DVR:v10.02.09.400-k
QLE2872 FW:v9.15.06 DVR:v10.02.09.400-k
```

2. Verify that `ql2xnvmeenable` is set. This enables the Marvell adapter to function as an NVMe/FC initiator:

```
cat /sys/module/qla2xxx/parameters/ql2xnvmeenable
```

   The expected output is 1.

NVMe/TCP

The NVMe/TCP protocol doesn't support the auto-connect operation. Instead, you can discover the NVMe/TCP subsystems and namespaces by performing the NVMe/TCP `connect` or `connect-all` operations manually.

Steps

1. Check that the initiator port can get the discovery log page data across the supported NVMe/TCP LIFs:

```
nvme discover -t tcp -w host-traddr -a traddr
```

Show example

```
nvme discover -t tcp -w 192.168.20.21 -a 192.168.20.28
Discovery Log Number of Records 8, Generation counter 10
=====Discovery Log Entry 0======
trtype:  tcp
adrfam:  ipv4
subtype: current discovery subsystem
treq:    not specified
portid:  8
trsvcid: 8009
subnqn:  nqn.1992-
08.com.netapp:sn.17e32b6e8c7f11f09545d039eac03c33:discovery
traddr:  192.168.21.29
eflags:  explicit discovery connections, duplicate discovery
information
sectype: none
=====Discovery Log Entry 1======
trtype:  tcp
adrfam:  ipv4
subtype: current discovery subsystem
treq:    not specified
portid:  6
trsvcid: 8009
subnqn:  nqn.1992-
08.com.netapp:sn.17e32b6e8c7f11f09545d039eac03c33:discovery
traddr:  192.168.20.29
eflags:  explicit discovery connections, duplicate discovery
information
sectype: none
=====Discovery Log Entry 2======
trtype:  tcp
adrfam:  ipv4
subtype: current discovery subsystem
treq:    not specified
portid:  7
trsvcid: 8009
subnqn:  nqn.1992-
08.com.netapp:sn.17e32b6e8c7f11f09545d039eac03c33:discovery
traddr:  192.168.21.28
eflags:  explicit discovery connections, duplicate discovery
information
sectype: none
=====Discovery Log Entry 3======
trtype:  tcp
adrfam:  ipv4
```

```
subtype: current discovery subsystem
treq:    not specified
portid:  5
trsvcid: 8009
subnqn:  nqn.1992-
08.com.netapp:sn.17e32b6e8c7f11f09545d039eac03c33:discovery
traddr:  192.168.20.28
eflags:  explicit discovery connections, duplicate discovery
information
sectype: none
=====Discovery Log Entry 4======
trtype:  tcp
adrfam:  ipv4
subtype: nvme subsystem
treq:    not specified
portid:  8
trsvcid: 4420
subnqn:  nqn.1992-
08.com.netapp:sn.17e32b6e8c7f11f09545d039eac03c33:subsystem.Bidi
rectional_DHCP_1_0
traddr:  192.168.21.29
eflags:  none
sectype: none
=====Discovery Log Entry 5======
trtype:  tcp
adrfam:  ipv4
subtype: nvme subsystem
treq:    not specified
portid:  6
trsvcid: 4420
subnqn:  nqn.1992-
08.com.netapp:sn.17e32b6e8c7f11f09545d039eac03c33:subsystem.Bidi
rectional_DHCP_1_0
traddr:  192.168.20.29
eflags:  none
sectype: none
=====Discovery Log Entry 6======
trtype:  tcp
adrfam:  ipv4
subtype: nvme subsystem
treq:    not specified
portid:  7
trsvcid: 4420
subnqn:  nqn.1992-
08.com.netapp:sn.17e32b6e8c7f11f09545d039eac03c33:subsystem.Bidi
rectional_DHCP_1_0
```

```
traddr:  192.168.21.28
eflags:  none
sectype: none
=====Discovery Log Entry 7======
trtype:  tcp
adrfam:  ipv4
subtype: nvme subsystem
treq:    not specified
portid:  5
trsvcid: 4420
subnqn:  nqn.1992-
08.com.netapp:sn.17e32b6e8c7f11f09545d039eac03c33:subsystem.Bidi
rectional_DHCP_1_0
traddr:  192.168.20.28
eflags:  none
sectype: non
```

2. Verify that the other NVMe/TCP initiator-target LIF combinations can successfully retrieve discovery log page data:

```
nvme discover -t tcp -w host-traddr -a traddr
```

Show example

```
nvme discover -t tcp -w 192.168.20.21 -a 192.168.20.28
nvme discover -t tcp -w 192.168.21.21 -a 192.168.21.28
nvme discover -t tcp -w 192.168.20.21 -a 192.168.20.29
nvme discover -t tcp -w 192.168.21.21 -a 192.168.21.29
```

3. Run the `nvme connect-all` command across all the supported NVMe/TCP initiator-target LIFs across the nodes:

```
nvme connect-all -t tcp -w host-traddr -a traddr
```

Show example

```
nvme    connect-all -t  tcp -w  192.168.20.21   -a 192.168.20.28
nvme    connect-all -t  tcp -w  192.168.21.21   -a 192.168.21.28
nvme    connect-all -t  tcp -w  192.168.20.21   -a 192.168.20.29
nvme    connect-all -t  tcp -w  192.168.21.21   -a 192.168.21.29
```

Beginning with RHEL 9.4, the setting for the NVMe/TCP `ctrl_loss_tmo timeout` is automatically set to "off". As a result:

- There are no limits on the number of retries (indefinite retry).
- You don't need to manually configure a specific `ctrl_loss_tmo timeout` duration when using the `nvme connect` or `nvme connect-all` commands (option -l).
- The NVMe/TCP controllers don't experience timeouts in the event of a path failure and remain connected indefinitely.

Step 4: Optionally, modify the iopolicy in the udev rules

RHEL 10.0 sets the default iopolicy for NVMe-oF to `round-robin`. If you are using RHEL 10.0 and want to change the iopolicy to `queue-depth`, modify the udev rules file as follows:

Steps

1. Open the udev rules file in a text editor with root privileges:

   ```
   /usr/lib/udev/rules.d/71-nvmf-netapp.rules
   ```

   You should see the following output:

   ```
   vi /usr/lib/udev/rules.d/71-nvmf-netapp.rules
   ```

2. Find the line that sets iopolicy for the NetApp ONTAP Controller, as shown in the following example rule:

   ```
   ACTION=="add", SUBSYSTEM=="nvme-subsystem", ATTR{subsystype}=="nvm",
   ATTR{model}=="NetApp ONTAP Controller", ATTR{iopolicy}="round-robin"
   ```

3. Modify the rule so that `round-robin` becomes `queue-depth`:

   ```
   ACTION=="add", SUBSYSTEM=="nvme-subsystem", ATTR{subsystype}=="nvm",
   ATTR{model}=="NetApp ONTAP Controller", ATTR{iopolicy}="queue-depth"
   ```

4. Reload the udev rules and apply the changes:

```
udevadm control --reload
udevadm trigger --subsystem-match=nvme-subsystem
```

5. Verify the current iopolicy for your subsystem. Replace <subsystem>, for example, `nvme-subsys0`.

```
cat /sys/class/nvme-subsystem/<subsystem>/iopolicy
```

You should see the following output:

```
queue-depth.
```

The new iopolicy applies automatically to matching NetApp ONTAP Controller devices. You don't need to reboot.

Step 5: Optionally, enable 1MB I/O for NVMe/FC

ONTAP reports a Max Data Transfer Size (MDTS) of 8 in the Identify Controller data. This means the maximum I/O request size can be up to 1MB. To issue I/O requests of size 1MB for a Broadcom NVMe/FC host, you should increase the `lpfc` value of the `lpfc_sg_seg_cnt` parameter to 256 from the default value of 64.

These steps don't apply to Qlogic NVMe/FC hosts.

Steps

1. Set the `lpfc_sg_seg_cnt` parameter to 256:

```
cat /etc/modprobe.d/lpfc.conf
```

You should see an output similar to the following example:

```
options lpfc lpfc_sg_seg_cnt=256
```

2. Run the `dracut -f` command, and reboot the host.
3. Verify that the value for `lpfc_sg_seg_cnt` is 256:

```
cat /sys/module/lpfc/parameters/lpfc_sg_seg_cnt
```

Step 6: Verify NVMe boot services

The `nvmefc-boot-connections.service` and `nvmf-autoconnect.service` boot services included in the NVMe/FC `nvme-cli` package are automatically enabled when the system boots.

After booting completes, verify that the `nvmefc-boot-connections.service` and `nvmf-autoconnect.service` boot services are enabled.

Steps

1. Verify that `nvmf-autoconnect.service` is enabled:

```
systemctl status nvmf-autoconnect.service
```

Show example output

```
nvmf-autoconnect.service - Connect NVMe-oF subsystems automatically
during boot
     Loaded: loaded (/usr/lib/systemd/system/nvmf-
autoconnect.service; enabled; preset: disabled)
     Active: inactive (dead) since Sun 2025-10-12 19:41:15 IST; 1
day 1h ago
 Invocation: 7b5b99929c6b41199d493fa25b629f6c
   Main PID: 10043 (code=exited, status=0/SUCCESS)
   Mem peak: 2.9M
        CPU: 50ms

Oct 12 19:41:15 localhost.localdomain systemd[1]: Starting nvmf-
autoconnect.service - Connect NVMe-oF subsystems automatically
during boot...
Oct 12 19:41:15 localhost.localdomain systemd[1]: nvmf-
autoconnect.service: Deactivated successfully.
Oct 12 19:41:15 localhost.localdomain systemd[1]: Finished nvmf-
autoconnect.service - Connect NVMe-oF subsystems automatically
during boot.
```

2. Verify that `nvmefc-boot-connections.service` is enabled:

```
systemctl status nvmefc-boot-connections.service
```

Show example output

```
nvmefc-boot-connections.service - Auto-connect to subsystems on FC-
NVME devices found during boot
     Loaded: loaded (/usr/lib/systemd/system/nvmefc-boot-
connections.service; enabled; preset: enabled)
     Active: inactive (dead) since Sun 2025-10-12 19:40:33 IST; 1
day 1h ago
 Invocation: 0ec258a9f8c342ffb82408086d409bc6
   Main PID: 4151 (code=exited, status=0/SUCCESS)
   Mem peak: 2.9M
        CPU: 17ms

Oct 12 19:40:33 localhost systemd[1]: Starting nvmefc-boot-
connections.service - Auto-connect to subsystems on FC-NVME devices
found during boot...
Oct 12 19:40:33 localhost systemd[1]: nvmefc-boot-
connections.service: Deactivated successfully.
Oct 12 19:40:33 localhost systemd[1]: Finished nvmefc-boot-
connections.service - Auto-connect to subsystems on FC-NVME devices
found during boot.
```

Step 7: Verify the multipathing configuration

Verify that the in-kernel NVMe multipath status, ANA status, and ONTAP namespaces are correct for the NVMe-oF configuration.

Steps

1. Verify that the appropriate NVMe-oF settings (such as, model set to NetApp ONTAP Controller and load balancing iopolicy set to queue-depth) for the respective ONTAP namespaces correctly display on the host:
   a. Display the subsystems:

      ```
      cat /sys/class/nvme-subsystem/nvme-subsys*/model
      ```

      You should see the following output:

      ```
      NetApp ONTAP Controller
      NetApp ONTAP Controller
      ```

   b. Display the policy:

      ```
      cat /sys/class/nvme-subsystem/nvme-subsys*/iopolicy
      ```

You should see the following output:

```
queue-depth
queue-depth
```

2. Verify that the namespaces are created and correctly discovered on the host:

```
nvme list
```

Show example

```
Node                  Generic               SN
Model
--------------------- --------------------- --------------------
----------------------------------------
/dev/nvme11n1         /dev/ng11n1           81OcqJXhgWtsAAAAAAAI
NetApp ONTAP Controller

Namespace  Usage                      Format           FW Rev
---------- ------------------------- ---------------- --------
0x1        951.90  MB /  21.47  GB    4 KiB +  0 B     9.18.1
```

3. Verify that the controller state of each path is live and has the correct ANA status:

NVMe/FC

```
nvme list-subsys /dev/nvme9n2
```

Show example

```
nvme-subsys9 - NQN=nqn.1992-
08.com.netapp:sn.7c34ab26675e11f0a6c0d039eac03c33:subsystem.subs
ystem_46
               hostnqn=nqn.2014-08.org.nvmexpress:uuid:4c4c4544-
0056-5410-8048-c7c04f425633
\
 +- nvme105 fc traddr=nn-0x2018d039eac03c33:pn-
0x201bd039eac03c33,host_traddr=nn-0x2000f4c7aa0cd7c3:pn-
0x2100f4c7aa0cd7c3 live optimized
 +- nvme107 fc traddr=nn-0x2018d039eac03c33:pn-
0x2019d039eac03c33,host_traddr=nn-0x2000f4c7aa0cd7c2:pn-
0x2100f4c7aa0cd7c2 live optimized
 +- nvme42 fc traddr=nn-0x2018d039eac03c33:pn-
0x201cd039eac03c33,host_traddr=nn-0x2000f4c7aa0cd7c3:pn-
0x2100f4c7aa0cd7c3 live optimized
 +- nvme44 fc traddr=nn-0x2018d039eac03c33:pn-
0x201ad039eac03c33,host_traddr=nn-0x2000f4c7aa0cd7c2:pn-
0x2100f4c7aa0cd7c2 live optimized
```

NVMe/TCP

```
nvme list-subsys /dev/nvme4n2
```

Show example

```
nvme-subsys4 - NQN=nqn.1992-
08.com.netapp:sn.17e32b6e8c7f11f09545d039eac03c33:subsystem.Bidi
rectional_DHCP_1_0
               hostnqn=nqn.2014-08.org.nvmexpress:uuid:4c4c4544-
0054-5110-8039-c3c04f523034
\
 +- nvme4 tcp
traddr=192.168.20.28,trsvcid=4420,host_traddr=192.168.20.21,src_
addr=192.168.20.21 live optimized
 +- nvme5 tcp
traddr=192.168.20.29,trsvcid=4420,host_traddr=192.168.20.21,src_
addr=192.168.20.21 live optimized
 +- nvme6 tcp
traddr=192.168.21.28,trsvcid=4420,host_traddr=192.168.21.21,src_
addr=192.168.21.21 live optimized
 +- nvme7 tcp
traddr=192.168.21.29,trsvcid=4420,host_traddr=192.168.21.21,src_
addr=192.168.21.21 live optimized
```

4. Verify that the NetApp plug-in displays the correct values for each ONTAP namespace device:

Column

```
nvme netapp ontapdevices -o column
```

Show example

```
Device           Vserver                   Subsystem
Namespace Path
---------------- ------------------------
------------------------ ------------------
/dev/nvme0n1     vs_nvme_sanboot_tcp       rhel_sanboot_tcp170
tcp_97

NSID UUID                                   Size
---- ---------------------------------------- ---------
1    982c0f2a-6b8b-11f0-a6c0-d039eac03c33   322.12GB
```

JSON

```
nvme netapp ontapdevices -o json
```

Show example

```
{
  "ONTAPdevices":[
    {
      "Device":"/dev/nvme0n1",
      "Vserver":"vs_nvme_sanboot_tcp",
      "Subsystem":"rhel_sanboot_tcp170",
      "Namespace_Path":"tcp_97",
      "NSID":1,
      "UUID":"982c0f2a-6b8b-11f0-a6c0-d039eac03c33",
      "LBA_Size":4096,
      "Namespace_Size":322122547200,
      "UsedBytes":16285069312,
      "Version":"9.18.1"
    }
]
}
```

Step 8: Set up secure in-band authentication

Secure in-band authentication is supported over NVMe/TCP between a RHEL 10.x host and an ONTAP controller.

Each host or controller must be associated with a `DH-HMAC-CHAP` key to set up secure authentication. A `DH-HMAC-CHAP` key is a combination of the NQN of the NVMe host or controller and an authentication secret configured by the administrator. To authenticate its peer, an NVMe host or controller must recognize the key associated with the peer.

Steps

Set up secure in-band authentication using the CLI or a config JSON file. If you need to specify different dhchap keys for different subsystems, you must use a config JSON file.

CLI

Set up secure in-band authentication using the CLI.

1. Obtain the host NQN:

```
cat /etc/nvme/hostnqn
```

2. Generate the dhchap key for the RHEL 10.x host.

   The following output describes the `gen-dhchap-key` command parameters:

```
nvme gen-dhchap-key -s optional_secret -l key_length {32|48|64} -m
HMAC_function {0|1|2|3} -n host_nqn
•   -s secret key in hexadecimal characters to be used to initialize
the host key
•   -l length of the resulting key in bytes
•   -m HMAC function to use for key transformation
0 = none, 1- SHA-256, 2 = SHA-384, 3=SHA-512
•   -n host NQN to use for key transformation
```

   In the following example, a random dhchap key with HMAC set to 3 (SHA-512) is generated.

```
nvme gen-dhchap-key -m 3 -n nqn.2014-
08.org.nvmexpress:uuid:4c4c4544-0054-5110-8039-c3c04f523034
DHHC-
1:03:AppJHkJygA6ZC4BxyQNtJST+4k4IOv47MAJk0xBITwFOHIC2nV/uE04RoSpy1z2
SXYqNW1bhLe9hJ+MDHigGexaG2Ig=:
```

3. On the ONTAP controller, add the host and specify both dhchap keys:

```
vserver nvme subsystem host add -vserver <svm_name> -subsystem
<subsystem> -host-nqn <host_nqn> -dhchap-host-secret
<authentication_host_secret> -dhchap-controller-secret
<authentication_controller_secret> -dhchap-hash-function {sha-
256|sha-512} -dhchap-group {none|2048-bit|3072-bit|4096-bit|6144-
bit|8192-bit}
```

4. A host supports two types of authentication methods, unidirectional and bidirectional. On the host, connect to the ONTAP controller and specify dhchap keys based on the chosen authentication method:

```
nvme connect -t tcp -w <host-traddr> -a <tr-addr> -n <host_nqn> -S
<authentication_host_secret> -C <authentication_controller_secret>
```

5. Validate the `nvme connect authentication` command by verifying the host and controller dhchap keys:

   a. Verify the host dhchap keys:

   ```
   cat /sys/class/nvme-subsystem/<nvme-subsysX>/nvme*/dhchap_secret
   ```

   Show example output for a unidirectional configuration

   ```
   cat /sys/class/nvme-subsystem/nvme-subsys4/nvme*/dhchap_secret
   DHHC-1:01:2G7lsg9PMO00h1Wf1g4QtP0XT11kREz0qVuLm2xvZdbaWR/g:
   DHHC-1:01:2G7lsg9PMO00h1Wf1g4QtP0XT11kREz0qVuLm2xvZdbaWR/g:
   DHHC-1:01:2G7lsg9PMO00h1Wf1g4QtP0XT11kREz0qVuLm2xvZdbaWR/g:
   DHHC-1:01:2G7lsg9PMO00h1Wf1g4QtP0XT11kREz0qVuLm2xvZdbaWR/g:
   ```

   b. Verify the controller dhchap keys:

   ```
   cat /sys/class/nvme-subsystem/<nvme-
   subsysX>/nvme*/dhchap_ctrl_secret
   ```

   Show example output for a bidirectional configuration

   ```
   cat /sys/class/nvme-subsystem/nvme-
   subsys4/nvme*/dhchap_ctrl_secret
   DHHC-
   1:03:5CgWULVnU5HUOwP1MNg95pkiUAwayiO+IvrALZR8HpeJIHw3xyHdGlTnv
   EJ81HDjBb+fGteUgIn0fj8ASHZIgkuFIx8=:
   DHHC-
   1:03:5CgWULVnU5HUOwP1MNg95pkiUAwayiO+IvrALZR8HpeJIHw3xyHdGlTnv
   EJ81HDjBb+fGteUgIn0fj8ASHZIgkuFIx8=:
   DHHC-
   1:03:5CgWULVnU5HUOwP1MNg95pkiUAwayiO+IvrALZR8HpeJIHw3xyHdGlTnv
   EJ81HDjBb+fGteUgIn0fj8ASHZIgkuFIx8=:
   DHHC-
   1:03:5CgWULVnU5HUOwP1MNg95pkiUAwayiO+IvrALZR8HpeJIHw3xyHdGlTnv
   EJ81HDjBb+fGteUgIn0fj8ASHZIgkuFIx8=:
   ```

JSON

When multiple NVMe subsystems are available on the ONTAP controller, you can use the `/etc/nvme/config.json` file with the `nvme connect-all` command.

Use the `-o` option to generate the JSON file. Refer to the NVMe connect-all man pages for more syntax options.

1. Configure the JSON file.

In the following example, `dhchap_key` corresponds to `dhchap_secret` and `dhchap_ctrl_key` corresponds to `dhchap_ctrl_secret`.

Show example


```
[
  {
    "hostnqn":"nqn.2014-08.org.nvmexpress:uuid:4c4c4544-0054-
5110-8039-c3c04f523034",
    "hostid":"44454c4c-5400-1051-8039-c3c04f523034",
    "dhchap_key":"DHHC-
1:01:2G7lsg9PMO00h1Wf1g4QtP0XT11kREz0qVuLm2xvZdbaWR/g:",
    "subsystems":[
      {
        "nqn":"nqn.1992-
08.com.netapp:sn.5857c8c9b22411f08d0ed039eac03c33:subsystem.Bidi
rectional_DHCP_1_0",
        "ports":[
          {
            "transport":"tcp",
            "traddr":"192.168.20.28",
            "host_traddr":"192.168.20.21",
            "trsvcid":"4420",
            "dhchap_ctrl_key":"DHHC-
1:03:5CgWULVnU5HUOwP1MNg95pkiUAwayiO+IvrALZR8HpeJIHw3xyHdGlTnvEJ
81HDjBb+fGteUgIn0fj8ASHZIgkuFIx8=:"
          },
          {
            "transport":"tcp",
            "traddr":"192.168.20.29",
            "host_traddr":"192.168.20.21",
            "trsvcid":"4420",
            "dhchap_ctrl_key":"DHHC-
1:03:5CgWULVnU5HUOwP1MNg95pkiUAwayiO+IvrALZR8HpeJIHw3xyHdGlTnvEJ
81HDjBb+fGteUgIn0fj8ASHZIgkuFIx8=:"
          },
          {
            "transport":"tcp",
            "traddr":"192.168.21.28",
            "host_traddr":"192.168.21.21",
            "trsvcid":"4420",
            "dhchap_ctrl_key":"DHHC-
1:03:5CgWULVnU5HUOwP1MNg95pkiUAwayiO+IvrALZR8HpeJIHw3xyHdGlTnvEJ
81HDjBb+fGteUgIn0fj8ASHZIgkuFIx8=:"
          },
          {
            "transport":"tcp",
            "traddr":"192.168.21.29",
            "host_traddr":"192.168.21.21",
```

```
            "trsvcid":"4420",
            "dhchap_ctrl_key":"DHHC-
1:03:5CgWULVnU5HUOwP1MNg95pkiUAwayiO+IvrALZR8HpeJIHw3xyHdGlTnvEJ
81HDjBb+fGteUgIn0fj8ASHZIgkuFIx8=:"
          }
        ]
      }
    ]
  }
]
```

2. Connect to the ONTAP controller using the config JSON file:

```
nvme connect-all -J /etc/nvme/config.json
```

Show example

```
traddr=192.168.20.28 is already connected
traddr=192.168.20.28 is already connected
traddr=192.168.20.29 is already connected
traddr=192.168.20.29 is already connected
```

3. Verify that the dhchap secrets have been enabled for the respective controllers for each subsystem.
   a. Verify the host dhchap keys:

```
cat /sys/class/nvme-subsystem/nvme-subsys4/nvme4/dhchap_secret
```

   The following example shows a dhchap key:

```
DHHC-1:01:2G7lsg9PMO00h1Wf1g4QtP0XT11kREz0qVuLm2xvZdbaWR/g:
```

   b. Verify the controller dhchap keys:

```
cat /sys/class/nvme-subsystem/nvme-
subsys4/nvme4/dhchap_ctrl_secret
```

   You should see an output similar to the following example:

```
DHHC-
1:03:5CgWULVnU5HUOwP1MNg95pkiUAwayiO+IvrALZR8HpeJIHw3xyHdGlTnvEJ8
1HDjBb+fGteUgIn0fj8ASHZIgkuFIx8=:
```

Step 9: Review the known issues

There are no known issues.

Configure RHEL 9.x for NVMe-oF with ONTAP storage

Red Hat Enterpirse Linux (RHEL) hosts support the NVMe over Fibre Channel (NVMe/FC) and NVMe over TCP (NVMe/TCP) protocols with Asymmetric Namespace Access (ANA). ANA provides multipathing functionality equivalent to asymmetric logical unit access (ALUA) in iSCSI and FCP environments.

Learn how to configure NVMe over Fabrics (NVMe-oF) hosts for RHEL 9.x. For more support and feature information, see RHEL ONTAP support and features.

NVMe-oF with RHEL 9.x has the following known limitations:

- The `nvme disconnect-all` command disconnects both root and data filesystems and might lead to system instability. Do not issue this on systems booting from SAN over NVMe-TCP or NVMe-FC namespaces.

Step 1: Optionally, enable SAN booting

You can configure your host to use SAN booting to simplify deployment and improve scalability. Use the Interoperability Matrix Tool to verify that your Linux OS, host bus adapter (HBA), HBA firmware, HBA boot BIOS, and ONTAP version support SAN booting.

Steps

1. Create a NVMe namespace and map it to the host.
2. Enable SAN booting in the server BIOS for the ports to which the SAN boot namespace is mapped.

   For information on how to enable the HBA BIOS, see your vendor-specific documentation.

3. Reboot the host and verify that the OS is up and running.

Step 2: Install RHEL and NVMe software and verify your configuration

To configure your host for NVMe-oF you need to install the host and NVMe software packages, enable multipathing, and verify your host NQN configuration.

Steps

1. Install RHEL 9.x on the server. After the installation is complete, verify that you are running the required RHEL 9.x kernel:

```
uname -r
```

Example RHEL kernel version:

```
5.14.0-611.5.1.el9_7.x86_64
```

2. Install the `nvme-cli` package:

```
rpm -qa|grep nvme-cli
```

The following example shows an `nvme-cli` package version:

```
nvme-cli-2.13-1.el9.x86_64
```

3. Install the `libnvme` package:

```
rpm -qa|grep libnvme
```

The following example shows an `libnvme` package version:

```
libnvme-1.13-1.el9.x86_64
```

4. On the host, check the hostnqn string at `/etc/nvme/hostnqn`:

```
cat /etc/nvme/hostnqn
```

The following example shows an `hostnqn` version:

```
nqn.2014-08.org.nvmexpress:uuid:4c4c4544-0056-5410-8048-b9c04f425633
```

5. On the ONTAP system, verify that the `hostnqn` string matches the `hostnqn` string for the corresponding subsystem on the ONTAP storage system:

```
::> vserver nvme subsystem host show -vserver vs_188
```

Show example

```
Vserver Subsystem Priority  Host NQN
------- --------- --------
-------------------------------------------------
vs_188  Nvme1
                  regular   nqn.2014-
08.org.nvmexpress:uuid:4c4c4544-0056-5410-8048-b9c04f425633
        Nvme10
                  regular   nqn.2014-
08.org.nvmexpress:uuid:4c4c4544-0056-5410-8048-b9c04f425633
        Nvme11
                  regular   nqn.2014-
08.org.nvmexpress:uuid:4c4c4544-0056-5410-8048-b9c04f425633
        Nvme12
                  regular   nqn.2014-
08.org.nvmexpress:uuid:4c4c4544-0056-5410-8048-b9c04f425633
48 entries were displayed.
```

If the `hostnqn` strings do not match, use the `vserver modify` command to update the `hostnqn` string on your corresponding ONTAP storage system subsystem to match the `hostnqn` string from `/etc/nvme/hostnqn` on the host.

Step 3: Configure NVMe/FC and NVMe/TCP

Configure NVMe/FC with Broadcom/Emulex or Marvell/QLogic adapters, or configure NVMe/TCP using manual discovery and connect operations.

NVMe/FC - Broadcom/Emulex

Configure NVMe/FC for a Broadcom/Emulex adapter.

Steps

1. Verify that you are using the supported adapter model:
   a. Display the model names:

```
cat /sys/class/scsi_host/host*/modelname
```

   You should see the following output:

```
LPe36002-M64
LPe36002-M64
```

   b. Display the model descriptions:

```
cat /sys/class/scsi_host/host*/modeldesc
```

   You should see an output similar to the following example:

```
Emulex LightPulse LPe36002-M64 2-Port 64Gb Fibre Channel Adapter
Emulex LightPulse LPe36002-M64 2-Port 64Gb Fibre Channel Adapter
```

2. Verify that you are using the recommended Broadcom `lpfc` firmware and inbox driver:
   a. Display the firmware version:

```
cat /sys/class/scsi_host/host*/fwrev
```

   The command returns the firmware versions:

```
14.4.393.53, sli-4:6:d
14.4.393.53, sli-4:6:d
```

   b. Display the inbox driver version:

```
cat /sys/module/lpfc/version
```

   The following example shows a driver version:

```
0:14.4.0.9
```

For the current list of supported adapter driver and firmware versions, see the Interoperability Matrix Tool.

3. Verify that `lpfc_enable_fc4_type` is set to `3`:

```
cat /sys/module/lpfc/parameters/lpfc_enable_fc4_type
```

4. Verify that you can view your initiator ports:

```
cat /sys/class/fc_host/host*/port_name
```

The following example shows port identities:

```
0x100000109bf044b1
0x100000109bf044b2
```

5. Verify that your initiator ports are online:

```
cat /sys/class/fc_host/host*/port_state
```

You should see the following output:

```
Online
Online
```

6. Verify that the NVMe/FC initiator ports are enabled and that the target ports are visible:

```
cat /sys/class/scsi_host/host*/nvme_info
```

Show example

```
NVME Initiator Enabled
XRI Dist lpfc0 Total 6144 IO 5894 ELS 250
NVME LPORT lpfc0 WWPN x100000109b954518 WWNN x200000109b954518
DID x020700 ONLINE
NVME RPORT       WWPN x2022d039eaa7dfc8 WWNN x201fd039eaa7dfc8
DID x020b03 TARGET DISCSRVC ONLINE
NVME RPORT       WWPN x2023d039eaa7dfc8 WWNN x201fd039eaa7dfc8
DID x020103 TARGET DISCSRVC ONLINE

NVME Statistics
LS: Xmt 0000000548 Cmpl 0000000548 Abort 00000000
LS XMIT: Err 00000000  CMPL: xb 00000000 Err 00000000
Total FCP Cmpl 0000000000001a68 Issue 0000000000001a68 OutIO
0000000000000000
        abort 00000000 noxri 00000000 nondlp 00000000 qdepth
00000000 wqerr 00000000 err 00000000
FCP CMPL: xb 00000000 Err 00000000

NVME Initiator Enabled
XRI Dist lpfc1 Total 6144 IO 5894 ELS 250
NVME LPORT lpfc1 WWPN x100000109b954519 WWNN x200000109b954519
DID x020500 ONLINE
NVME RPORT       WWPN x2027d039eaa7dfc8 WWNN x2025d039eaa7dfc8
DID x020b01 TARGET DISCSRVC ONLINE

NVME Statistics
LS: Xmt 00000005ab Cmpl 00000005ab Abort 00000000
LS XMIT: Err 00000000  CMPL: xb 00000000 Err 00000000
Total FCP Cmpl 0000000000086ce1 Issue 0000000000086ce2 OutIO
0000000000000001
        abort 0000009c noxri 00000000 nondlp 00000002 qdepth
00000000 wqerr 00000000 err 00000000
FCP CMPL: xb 000000b8 Err 000000b8

NVME Initiator Enabled
XRI Dist lpfc2 Total 6144 IO 5894 ELS 250
NVME LPORT lpfc2 WWPN x100000109bf044b1 WWNN x200000109bf044b1
DID x022a00 ONLINE
NVME RPORT       WWPN x2027d039eaa7dfc8 WWNN x2025d039eaa7dfc8
DID x020b01 TARGET DISCSRVC ONLINE
NVME RPORT       WWPN x2011d039eaa7dfc8 WWNN x200fd039eaa7dfc8
DID x020b02 TARGET DISCSRVC ONLINE
NVME RPORT       WWPN x2002d039eaa7dfc8 WWNN x2000d039eaa7dfc8
DID x020b05 TARGET DISCSRVC ONLINE
```

```
NVME RPORT       WWPN x2026d039eaa7dfc8 WWNN x2025d039eaa7dfc8
DID x021301 TARGET DISCSRVC ONLINE
NVME RPORT       WWPN x2010d039eaa7dfc8 WWNN x200fd039eaa7dfc8
DID x021302 TARGET DISCSRVC ONLINE
NVME RPORT       WWPN x2001d039eaa7dfc8 WWNN x2000d039eaa7dfc8
DID x021305 TARGET DISCSRVC ONLINE

NVME Statistics
LS: Xmt 000000c186 Cmpl 000000c186 Abort 00000000
LS XMIT: Err 00000000  CMPL: xb 00000000 Err 00000000
Total FCP Cmpl 00000000c348ca37 Issue 00000000c3344057 OutIO
fffffffffeb7620
        abort 0000815b noxri 000018b5 nondlp 00000116 qdepth
00000000 wqerr 00000000 err 00000000
FCP CMPL: xb 0000915b Err 000c6091

NVME Initiator Enabled
XRI Dist lpfc3 Total 6144 IO 5894 ELS 250
NVME LPORT lpfc3 WWPN x100000109bf044b2 WWNN x200000109bf044b2
DID x021b00 ONLINE
NVME RPORT       WWPN x2028d039eaa7dfc8 WWNN x2025d039eaa7dfc8
DID x020101 TARGET DISCSRVC ONLINE
NVME RPORT       WWPN x2012d039eaa7dfc8 WWNN x200fd039eaa7dfc8
DID x020102 TARGET DISCSRVC ONLINE
NVME RPORT       WWPN x2003d039eaa7dfc8 WWNN x2000d039eaa7dfc8
DID x020105 TARGET DISCSRVC ONLINE
NVME RPORT       WWPN x2029d039eaa7dfc8 WWNN x2025d039eaa7dfc8
DID x022901 TARGET DISCSRVC ONLINE
NVME RPORT       WWPN x2013d039eaa7dfc8 WWNN x200fd039eaa7dfc8
DID x022902 TARGET DISCSRVC ONLINE
NVME RPORT       WWPN x2004d039eaa7dfc8 WWNN x2000d039eaa7dfc8
DID x022905 TARGET DISCSRVC ONLINE

NVME Statistics
LS: Xmt 000000c186 Cmpl 000000c186 Abort 00000000
LS XMIT: Err 00000000  CMPL: xb 00000000 Err 00000000
Total FCP Cmpl 00000000b5761af5 Issue 00000000b564b55e OutIO
fffffffffee9a69
        abort 000083d7 noxri 000016ea nondlp 00000195 qdepth
00000000 wqerr 00000002 err 00000000
FCP CMPL: xb 000094a4 Err 000c22e7
```

NVMe/FC - Marvell/QLogic

Configure NVMe/FC for a Marvell/QLogic adapter.

Steps

1. Verify that you are using the supported adapter driver and firmware versions:

```
cat /sys/class/fc_host/host*/symbolic_name
```

   The following example shows driver and firmware versions:

```
QLE2872 FW:v9.15.06 DVR:v10.02.09.400-k
QLE2872 FW:v9.15.06 DVR:v10.02.09.400-k
```

2. Verify that `ql2xnvmeenable` is set. This enables the Marvell adapter to function as an NVMe/FC initiator:

```
cat /sys/module/qla2xxx/parameters/ql2xnvmeenable
```

   The expected output is 1.

NVMe/TCP

The NVMe/TCP protocol doesn't support the auto-connect operation. Instead, you can discover the NVMe/TCP subsystems and namespaces by performing the NVMe/TCP `connect` or `connect-all` operations manually.

Steps

1. Check that the initiator port can get the discovery log page data across the supported NVMe/TCP LIFs:

```
nvme discover -t tcp -w host-traddr -a traddr
```

Show example

```
nvme discover -t tcp -w 192.168.30.15 -a 192.168.30.48

Discovery Log Number of Records 8, Generation counter 18
=====Discovery Log Entry 0======
trtype:  tcp
adrfam:  ipv4
subtype: current discovery subsystem
treq:    not specified
portid:  8
trsvcid: 8009
subnqn:  nqn.1992-
08.com.netapp:sn.51a3c9846e0c11f08f5dd039eaa7dfc9:discovery
traddr:  192.168.31.49
eflags:  explicit discovery connections, duplicate discovery
information
sectype: none
=====Discovery Log Entry 1======
trtype:  tcp
adrfam:  ipv4
subtype: current discovery subsystem
treq:    not specified
portid:  7
trsvcid: 8009
subnqn:  nqn.1992-
08.com.netapp:sn.51a3c9846e0c11f08f5dd039eaa7dfc9:discovery
traddr:  192.168.31.48
eflags:  explicit discovery connections, duplicate discovery
information
sectype: none
=====Discovery Log Entry 2======
trtype:  tcp
adrfam:  ipv4
subtype: current discovery subsystem
treq:    not specified
portid:  6
trsvcid: 8009
subnqn:  nqn.1992-
08.com.netapp:sn.51a3c9846e0c11f08f5dd039eaa7dfc9:discovery
traddr:  192.168.30.49
eflags:  explicit discovery connections, duplicate discovery
information
sectype: none
=====Discovery Log Entry 3======
trtype:  tcp
```

```
adrfam:  ipv4
subtype: current discovery subsystem
treq:    not specified
portid:  5
trsvcid: 8009
subnqn:  nqn.1992-
08.com.netapp:sn.51a3c9846e0c11f08f5dd039eaa7dfc9:discovery
traddr:  192.168.30.48
eflags:  explicit discovery connections, duplicate discovery
information
sectype: none
=====Discovery Log Entry 4======
trtype:  tcp
adrfam:  ipv4
subtype: nvme subsystem
treq:    not specified
portid:  8
trsvcid: 4420
subnqn:  nqn.1992-
08.com.netapp:sn.51a3c9846e0c11f08f5dd039eaa7dfc9:subsystem.Nvme
38
traddr:  192.168.31.49
eflags:  none
sectype: none
=====Discovery Log Entry 5======
trtype:  tcp
adrfam:  ipv4
subtype: nvme subsystem
treq:    not specified
portid:  7
trsvcid: 4420
subnqn:  nqn.1992-
08.com.netapp:sn.51a3c9846e0c11f08f5dd039eaa7dfc9:subsystem.Nvme
38
traddr:  192.168.31.48
eflags:  none
sectype: none
=====Discovery Log Entry 6======
trtype:  tcp
adrfam:  ipv4
subtype: nvme subsystem
treq:    not specified
portid:  6
trsvcid: 4420
subnqn:  nqn.1992-
08.com.netapp:sn.51a3c9846e0c11f08f5dd039eaa7dfc9:subsystem.Nvme
```

```
38
traddr:  192.168.30.49
eflags:  none
sectype: none
=====Discovery Log Entry 7======
trtype:  tcp
adrfam:  ipv4
subtype: nvme subsystem
treq:    not specified
portid:  5
trsvcid: 4420
subnqn:  nqn.1992-
08.com.netapp:sn.51a3c9846e0c11f08f5dd039eaa7dfc9:subsystem.Nvme
38
traddr:  192.168.30.48
eflags:  none
sectype: none
```

2. Verify that the other NVMe/TCP initiator-target LIF combinations can successfully retrieve discovery log page data:

```
nvme discover -t tcp -w host-traddr -a traddr
```

Show example

```
nvme discover -t tcp -w 192.168.30.15 -a 192.168.30.48
nvme discover -t tcp -w 192.168.30.15 -a 192.168.30.49
nvme discover -t tcp -w 192.168.31.15 -a 192.168.31.48
nvme discover -t tcp -w 192.168.31.15 -a 192.168.31.49
```

3. Run the `nvme connect-all` command across all the supported NVMe/TCP initiator-target LIFs across the nodes:

```
nvme connect-all -t tcp -w host-traddr -a traddr
```

Show example

```
nvme  connect-all -t  tcp -w  192.168.30.15 -a  192.168.30.48
nvme    connect-all -t  tcp -w  192.168.30.15   -a
192.168.30.49
nvme    connect-all -t  tcp -w  192.168.31.15   -a
192.168.31.48
nvme    connect-all -t  tcp -w  192.168.31.15   -a
192.168.31.49
```

Beginning with RHEL 9.4, the setting for the NVMe/TCP `ctrl_loss_tmo timeout` is automatically set to "off". As a result:

- There are no limits on the number of retries (indefinite retry).
- You don't need to manually configure a specific `ctrl_loss_tmo timeout` duration when using the `nvme connect` or `nvme connect-all` commands (option -l).
- The NVMe/TCP controllers don't experience timeouts in the event of a path failure and remain connected indefinitely.

Step 4: Optionally, modify the iopolicy in the udev rules

RHEL 9.6 sets the default iopolicy for NVMe-oF to `round-robin`. If you are using RHEL 9.6 and want to change the iopolicy to `queue-depth`, modify the udev rules file as follows:

Steps

1. Open the udev rules file in a text editor with root privileges:

   ```
   /usr/lib/udev/rules.d/71-nvmf-netapp.rules
   ```

   You should see the following output:

   ```
   vi /usr/lib/udev/rules.d/71-nvmf-netapp.rules
   ```

2. Find the line that sets iopolicy for the NetApp ONTAP Controller, as shown in the following example rule:

   ```
   ACTION=="add", SUBSYSTEM=="nvme-subsystem", ATTR{subsystype}=="nvm",
   ATTR{model}=="NetApp ONTAP Controller", ATTR{iopolicy}="round-robin"
   ```

3. Modify the rule so that `round-robin` becomes `queue-depth`:

```
ACTION=="add", SUBSYSTEM=="nvme-subsystem", ATTR{subsystype}=="nvm",
ATTR{model}=="NetApp ONTAP Controller", ATTR{iopolicy}="queue-depth"
```

4. Reload the udev rules and apply the changes:

```
udevadm control --reload
udevadm trigger --subsystem-match=nvme-subsystem
```

5. Verify the current iopolicy for your subsystem. Replace <subsystem>, for example, `nvme-subsys0`.

```
cat /sys/class/nvme-subsystem/<subsystem>/iopolicy
```

You should see the following output:

```
queue-depth.
```

The new iopolicy applies automatically to matching NetApp ONTAP Controller devices. You don't need to reboot.

Step 5: Optionally, enable 1MB I/O for NVMe/FC

ONTAP reports a Max Data Transfer Size (MDTS) of 8 in the Identify Controller data. This means the maximum I/O request size can be up to 1MB. To issue I/O requests of size 1MB for a Broadcom NVMe/FC host, you should increase the `lpfc` value of the `lpfc_sg_seg_cnt` parameter to 256 from the default value of 64.

These steps don't apply to Qlogic NVMe/FC hosts.

Steps

1. Set the `lpfc_sg_seg_cnt` parameter to 256:

```
cat /etc/modprobe.d/lpfc.conf
```

You should see an output similar to the following example:

```
options lpfc lpfc_sg_seg_cnt=256
```

2. Run the `dracut -f` command, and reboot the host.
3. Verify that the value for `lpfc_sg_seg_cnt` is 256:

```
cat /sys/module/lpfc/parameters/lpfc_sg_seg_cnt
```

Step 6: Verify NVMe boot services

The `nvmefc-boot-connections.service` and `nvmf-autoconnect.service` boot services included in the NVMe/FC `nvme-cli` package are automatically enabled when the system boots.

After booting completes, verify that the `nvmefc-boot-connections.service` and `nvmf-autoconnect.service` boot services are enabled.

Steps

1. Verify that `nvmf-autoconnect.service` is enabled:

```
systemctl status nvmf-autoconnect.service
```

Show example output

```
nvmf-autoconnect.service - Connect NVMe-oF subsystems automatically
during boot
        Loaded: loaded (/usr/lib/systemd/system/nvmf-
autoconnect.service;  enabled; preset: disabled)

Active: inactive (dead) since Wed 2025-10-29 00:42:03 EDT; 6h ago
Main PID: 8487 (code=exited, status=0/SUCCESS)  CPU: 66ms

Oct 29 00:42:03 R650-14-188 systemd[1]: Starting Connect NVMe-oF
subsystems automatically during boot...
Oct 29 00:42:03 R650-14-188 systemd[1]: nvmf-autoconnect.service:
Deactivated successfully.
Oct 29 00:42:03 R650-14-188 systemd[1]: Finished Connect NVMe-oF
subsystems automatically during boot.
```

2. Verify that `nvmefc-boot-connections.service` is enabled:

```
systemctl status nvmefc-boot-connections.service
```

Show example output

```
nvmefc-boot-connections.service - Auto-connect to subsystems on FC-
NVME devices found during boot
         Loaded: loaded (/usr/lib/systemd/system/nvmefc-boot-
connections.service; enabled; preset:enabled)
     Active: inactive (dead) since Wed 2025-10-29 00:41:51 EDT; 6h
ago
Main PID: 4652 (code=exited, status=0/SUCCESS)
        CPU: 13ms

Oct 29 00:41:51 R650-14-188 systemd[1]: Starting Auto-connect to
subsystems on FC-NVME devices found during boot...  Oct 29 00:41:51
R650-14-188 systemd[1]: nvmefc-boot-connections.service: Deactivated
successfully.  Oct 29 00:41:51 R650-14-188 systemd[1]: Finished
Auto-connect to subsystems on FC-NVME devices found during boot
```

Step 7: Verify the multipathing configuration

Verify that the in-kernel NVMe multipath status, ANA status, and ONTAP namespaces are correct for the NVMe-oF configuration.

Steps

1. Verify that the in-kernel NVMe multipath is enabled:

   ```
   cat /sys/module/nvme_core/parameters/multipath
   ```

   You should see the following output:

   ```
   Y
   ```

2. Verify that the appropriate NVMe-oF settings (such as, model set to NetApp ONTAP Controller and load balancing iopolicy set to round-robin) for the respective ONTAP namespaces correctly display on the host:

   a. Display the subsystems:

      ```
      cat /sys/class/nvme-subsystem/nvme-subsys*/model
      ```

      You should see the following output:

      ```
      NetApp ONTAP Controller
      NetApp ONTAP Controller
      ```

b. Display the policy:

```
cat /sys/class/nvme-subsystem/nvme-subsys*/iopolicy
```

You should see the following output:

```
queue-depth
queue-depth
```

3. Verify that the namespaces are created and correctly discovered on the host:

```
nvme list
```

Show example

```
Node                                  Generic              SN
Model
------------------------------------------------------------------------
------------------
/dev/nvme100n1  /dev/ng100n1  81LJCJYaKOHhAAAAAAAf   NetApp ONTAP
Controller
Namespace Usage    Format              FW              Rev
------------------------------------------------------------
0x1                 1.19  GB /   5.37  GB    4 KiB + 0 B   9.18.1
```

4. Verify that the controller state of each path is live and has the correct ANA status:

NVMe/FC

```
nvme list-subsys /dev/nvme100n1
```

Show example

```
nvme-subsys4 - NQN=nqn.1992-
08.com.netapp:sn.3623e199617311f09257d039eaa7dfc9:subsystem.Nvme
31
               hostnqn=nqn.2014-08.org.nvmexpress:uuid:
4c4c4544-0056-5410-8048-b9c04f42563
               \
+- nvme199 fc   traddr=nn-0x200fd039eaa7dfc8:pn-
0x2010d039eaa7dfc8,host_traddr=nn-0x200000109bf044b1:pn-
0x100000109bf044b1 live optimized
+- nvme246 fc  traddr=nn-0x200fd039eaa7dfc8:pn-
0x2011d039eaa7dfc8,host_traddr=nn-0x200000109bf044b1:pn-
0x100000109bf044b1  live non-optimized
+- nvme249 fc  traddr=nn-0x200fd039eaa7dfc8:pn-
0x2013d039eaa7dfc8,host_traddr=nn-0x200000109bf044b2:pn-
0x100000109bf044b2 live optimized
+- nvme251 fc   traddr=nn-0x200fd039eaa7dfc8:pn-
0x2012d039eaa7dfc8,host_traddr=nn-0x200000109bf044b2:pn-
0x100000109bf044b2 live non-optimized
```

NVMe/TCP

```
nvme list-subsys /dev/nvme0n1
```

Show example

```
nvme-subsys0 - NQN=nqn.1992-
08.com.netapp:sn.51a3c9846e0c11f08f5dd039eaa7dfc9:subsystem.Nvme
1
hostnqn=nqn.2014-08.org.nvmexpress:uuid:4c4c4544-0035-5910-804b-
b5c04f444d33
\
+- nvme0 tcp
traddr=192.168.30.48,trsvcid=4420,host_traddr=192.168.30.15,
src_addr=192.168.30.15 live optimized
+- nvme1 tcp
traddr=192.168.30.49,trsvcid=4420,host_traddr=192.168.30.15,
src_addr=192.168.30.15 live non-optimized
+- nvme2 tcp
traddr=192.168.31.48,trsvcid=4420,host_traddr=192.168.31.15,
src_addr=192.168.31.15 live optimized
+- nvme3 tcp
traddr=192.168.31.49,trsvcid=4420,host_traddr=192.168.31.15,
src_addr=192.168.31.15 live non-optimized
```

5. Verify that the NetApp plug-in displays the correct values for each ONTAP namespace device:

Column

```
nvme netapp ontapdevices -o column
```

Show example

```
Device          Vserver         Subsystem   Namespace Path
NSID
------------    --------        ----------- -----------------
----
/dev/nvme0n1    vs_iscsi_tcp    Nvme1       /vol/Nvmevol1/ns1
1
UUID                                        Size
--------------------------------------   -----
d8efef7d-4dde-447f-b50e-b2c009298c66    26.84GB
```

JSON

```
nvme netapp ontapdevices -o json
```

Show example

```
{
  "ONTAPdevices":[
    {
      "Device":"/dev/nvme0n1",
      "Vserver":"vs_iscsi_tcp",
      " Subsystem":"Nvme1",
      "Namespace_Path":"/vol/Nvmevol1/ns1",
      "NSID":1,
      "UUID":"d8efef7d-4dde-447f-b50e-b2c009298c66",
      "LBA_Size":4096,
      "Namespace_Size":26843545600,
    },
]
}
```

Step 8: Set up secure in-band authentication

Secure in-band authentication is supported over NVMe/TCP between a RHEL 9.x host and an ONTAP

controller.

Each host or controller must be associated with a `DH-HMAC-CHAP` key to set up secure authentication. A `DH-HMAC-CHAP` key is a combination of the NQN of the NVMe host or controller and an authentication secret configured by the administrator. To authenticate its peer, an NVMe host or controller must recognize the key associated with the peer.

Steps

Set up secure in-band authentication using the CLI or a config JSON file. If you need to specify different dhchap keys for different subsystems, you must use a config JSON file.

CLI

Set up secure in-band authentication using the CLI.

1. Obtain the host NQN:

```
cat /etc/nvme/hostnqn
```

2. Generate the dhchap key for the RHEL 9.x host.

   The following output describes the `gen-dhchap-key` command parameters:

```
nvme gen-dhchap-key -s optional_secret -l key_length {32|48|64} -m
HMAC_function {0|1|2|3} -n host_nqn
•   -s secret key in hexadecimal characters to be used to initialize
the host key
•   -l length of the resulting key in bytes
•   -m HMAC function to use for key transformation
0 = none, 1- SHA-256, 2 = SHA-384, 3=SHA-512
•   -n host NQN to use for key transformation
```

   In the following example, a random dhchap key with HMAC set to 3 (SHA-512) is generated.

```
nvme gen-dhchap-key -m 3 -n nqn.2014-
08.org.nvmexpress:uuid:e6dade64-216d-11ec-b7bb-7ed30a5482c3
DHHC-
1:03:wSpuuKbBHTzC0W9JZxMBsYd9JFV8Si9aDh22k2BR/4m852vH7KGlrJeMpzhmyjD
WOo0PJJM6yZsTeEpGkDHMHQ255+g=:
```

3. On the ONTAP controller, add the host and specify both dhchap keys:

```
vserver nvme subsystem host add -vserver <svm_name> -subsystem
<subsystem> -host-nqn <host_nqn> -dhchap-host-secret
<authentication_host_secret> -dhchap-controller-secret
<authentication_controller_secret> -dhchap-hash-function {sha-
256|sha-512} -dhchap-group {none|2048-bit|3072-bit|4096-bit|6144-
bit|8192-bit}
```

4. A host supports two types of authentication methods, unidirectional and bidirectional. On the host, connect to the ONTAP controller and specify dhchap keys based on the chosen authentication method:

```
nvme connect -t tcp -w <host-traddr> -a <tr-addr> -n <host_nqn> -S
<authentication_host_secret> -C <authentication_controller_secret>
```

5. Validate the `nvme connect authentication` command by verifying the host and controller dhchap keys:

   a. Verify the host dhchap keys:

   ```
   cat /sys/class/nvme-subsystem/<nvme-subsysX>/nvme*/dhchap_secret
   ```

   Show example output for a unidirectional configuration

   ```
   cat /sys/class/nvme-subsystem/nvme-subsys1/nvme*/dhchap_secret
   DHHC-1:01:hhdIYK7rGxHiNYS4d421GxHeDRUAuY0vmdqCp/NOaYND2PSc:
   DHHC-1:01:hhdIYK7rGxHiNYS4d421GxHeDRUAuY0vmdqCp/NOaYND2PSc:
   DHHC-1:01:hhdIYK7rGxHiNYS4d421GxHeDRUAuY0vmdqCp/NOaYND2PSc:
   DHHC-1:01:hhdIYK7rGxHiNYS4d421GxHeDRUAuY0vmdqCp/NOaYND2PSc:
   ```

   b. Verify the controller dhchap keys:

   ```
   cat /sys/class/nvme-subsystem/<nvme-
   subsysX>/nvme*/dhchap_ctrl_secret
   ```

   Show example output for a bidirectional configuration

   ```
   cat /sys/class/nvme-subsystem/nvme-
   subsys*/nvme*/dhchap_ctrl_secret

   DHHC-
   1:03:ZCRrP9MQOeXhFitT7Fvvf/3P6K/qY1HfSmSfM8nLjESJdOjbjK/J6m00y
   gJgjm0VrRlrgrnHzjtWJmsnoVBO3rPDGEk=:
   DHHC-
   1:03:ZCRrP9MQOeXhFitT7Fvvf/3P6K/qY1HfSmSfM8nLjESJdOjbjK/J6m00y
   gJgjm0VrRlrgrnHzjtWJmsnoVBO3rPDGEk=:
   DHHC-
   1:03:ZCRrP9MQOeXhFitT7Fvvf/3P6K/qY1HfSmSfM8nLjESJdOjbjK/J6m00y
   gJgjm0VrRlrgrnHzjtWJmsnoVBO3rPDGEk=:
   DHHC-
   1:03:ZCRrP9MQOeXhFitT7Fvvf/3P6K/qY1HfSmSfM8nLjESJdOjbjK/J6m00y
   gJgjm0VrRlrgrnHzjtWJmsnoVBO3rPDGEk=:
   ```

JSON

When multiple NVMe subsystems are available on the ONTAP controller, you can use the `/etc/nvme/config.json` file with the `nvme connect-all` command.

Use the `-o` option to generate the JSON file. Refer to the NVMe connect-all man pages for more syntax options.

1. Configure the JSON file.

In the following example, `dhchap_key` corresponds to `dhchap_secret` and `dhchap_ctrl_key` corresponds to `dhchap_ctrl_secret`.

Show example

```
cat /etc/nvme/config.json
[
{
  "hostnqn":"nqn.2014-08.org.nvmexpress:uuid:4c4c4544-0035-5910-
804b-b5c04f444d33",
  "hostid":"4c4c4544-0035-5910-804b-b5c04f444d33",
  "dhchap_key":"DHHC-
1:01:GhgaLS+0h0W/IxKhSa0iaMHg17SOHRTzBduPzoJ6LKEJs3/f:",
  "subsystems":[
        {
          "nqn":"nqn.1992-
08.com.netapp:sn.2c0c80d9873a11f0bc60d039eab6cb6d:subsystem.istp
MNTC_subsys",
          "ports":[
              {
                  "transport":"tcp",
                    "traddr":"192.168.30.44",
                  "host_traddr":"192.168.30.15",
                  "trsvcid":"4420",
                  "dhchap_ctrl_key":"DHHC-
1:03:GaraCO84o/uM0jF4rKJlgTy22bVoV0dRn1M+9QDfQRNVwJDHfPu2LrK5Y+/
XG8iGcRtBCdm3
fYm3ZmO6NiepCORoY5Q=:"
              },
              {
                  "transport":"tcp",
                  "traddr":"192.168.30.45"
                  "host_traddr":"192.168.30.15",
                  "trsvcid":"4420",
                  "dhchap_ctrl_key":"DHHC-
1:03:GaraCO84o/uM0jF4rKJlgTy22bVoV0dRn1M+9QDfQRNVwJDHfPu2LrK5Y+/
XG8iGcRtBCdm3
fYm3ZmO6NiepCORoY5Q=:"
              },
              {
                  "transport":"tcp",
                 "traddr":"192.168.31.44",
                  "host_traddr":"192.168.31.15",
                  "trsvcid":"4420",
                  "dhchap_ctrl_key":"DHHC-
                  1:03:
GaraCO84o/uM0jF4rKJlgTy22bVoV0dRn1M+9QDfQRNVwJDHfPu2LrK5Y+/XG8iG
c
RtBCdm3fYm3ZmO6NiepCORoY5Q=:"                              },
```

```
 {
                  "transport":"tcp",
                  "traddr":"192.168.31.45",
                  "host_traddr":"192.168.31.15",
                  "trsvcid":"4420",
                  "dhchap_ctrl_key":"DHHC-
                  1:03:
GaraCO84o/uM0jF4rKJlgTy22bVoV0dRn1M+9QDfQRNVwJDHfPu2LrK5Y+/XG8iG
cRtBCdm3fYm3ZmO6NiepCORoY5Q=:"
              }
        ]
  ]
}
]
```

2. Connect to the ONTAP controller using the config JSON file:

```
nvme connect-all -J /etc/nvme/config.json
```

Show example

```
already connected to hostnqn=nqn.2014-
08.org.nvmexpress:uuid:4c4c4544-0035-5910-804b-
b5c04f444d33,nqn=nqn.1992-
08.com.netapp:sn.2c0c80d9873a11f0bc60d039eab6cb6d:subsystem.istp
MNTC_subsys,transport=tcp,traddr=192.168.30.44,trsvcid=4420
already connected to hostnqn=nqn.2014-
08.org.nvmexpress:uuid:4c4c4544-0035-5910-804b-
b5c04f444d33,nqn=nqn.1992-
08.com.netapp:sn.2c0c80d9873a11f0bc60d039eab6cb6d:subsystem.istp
MNTC_subsys,transport=tcp,traddr=192.168.31.44,trsvcid=4420
already connected to hostnqn=nqn.2014-
08.org.nvmexpress:uuid:4c4c4544-0035-5910-804b-
b5c04f444d33,nqn=nqn.1992-
08.com.netapp:sn.2c0c80d9873a11f0bc60d039eab6cb6d:subsystem.istp
MNTC_subsys,transport=tcp,traddr=192.168.30.45,trsvcid=4420
already connected to hostnqn=nqn.2014-
08.org.nvmexpress:uuid:4c4c4544-0035-5910-804b-
b5c04f444d33,nqn=nqn.1992-
08.com.netapp:sn.2c0c80d9873a11f0bc60d039eab6cb6d:subsystem.istp
MNTC_subsys,transport=tcp,traddr=192.168.31.45,trsvcid=4420
```

3. Verify that the dhchap secrets have been enabled for the respective controllers for each subsystem:
   a. Verify the host dhchap keys:

```
cat /sys/class/nvme-subsystem/nvme-subsys96/nvme96/dhchap_secret
```

   The following example shows a dhchap key:

```
DHHC-1:01:hhdIYK7rGxHiNYS4d421GxHeDRUAuY0vmdqCp/NOaYND2PSc:
```

   b. Verify the controller dhchap keys:

```
cat /sys/class/nvme-subsystem/nvme-
subsys96/nvme96/dhchap_ctrl_secret
```

   You should see an output similar to the following example:

```
DHHC-
1:03:ZCRrP9MQOeXhFitT7Fvvf/3P6K/qY1HfSmSfM8nLjESJdOjbjK/J6m00ygJg
jm0VrRlrgrnHzjtWJmsnoVBO3rPDGEk=:
```

Step 9: Review the known issues

These are the known issues:

NetApp Bug ID	Title	Description
1503468	In RHEL 9.1, the `nvme list-subsys` command returns repeated nvme controller list for a given subsystem	The `nvme list-subsys` command returns a list of NVMe controllers for a given subsystem. In RHEL 9.1, this command shows controllers with their ANA state for all namespaces in the subsystem. Because ANA state is a per-namespace attribute, the command should display unique controller entries with the path state for each namespace.

NetApp Bug ID	Title	Description
1479047	RHEL 9.0 NVMe-oF hosts create duplicate Persistent Discovery Controllers (PDCs)	On NVMe-oF hosts, you can use the `nvme discover -p` command to create PDCs. When this command is used, only one PDC should be created per initiator-target combination. However, if you are running ONTAP 9.10.1 and RHEL 9.0 with an NVMe-oF host, a duplicate PDC is created each time `nvme discover -p` is executed. This leads to unnecessary usage of resources on both the host and the target.

Configure RHEL 8.x for NVMe-oF with ONTAP storage

Red Hat Enterpirse Linux (RHEL) hosts support the NVMe over Fibre Channel (NVMe/FC) and NVMe over TCP (NVMe/TCP) protocols with Asymmetric Namespace Access (ANA). ANA provides multipathing functionality equivalent to asymmetric logical unit access (ALUA) in iSCSI and FCP environments.

Learn how to configure NVMe over Fabrics (NVMe-oF) hosts for RHEL 8.x. For more support and feature information, see RHEL ONTAP support and features.

NVMe-oF with RHEL 8.x has the following known limitations:

- SAN booting using the NVMe-oF protocol is not currently supported.
- In-kernel NVMe multipath is disabled by default on NVMe-oF hosts in RHEL 8.x; you must enable it manually.
- NVMe/TCP is available as a technology preview due to known issues.

Step 1: Optionally, enable SAN booting

You can configure your host to use SAN booting to simplify deployment and improve scalability. Use the Interoperability Matrix Tool to verify that your Linux OS, host bus adapter (HBA), HBA firmware, HBA boot BIOS, and ONTAP version support SAN booting.

Steps

1. Create a NVMe namespace and map it to the host.
2. Enable SAN booting in the server BIOS for the ports to which the SAN boot namespace is mapped.

   For information on how to enable the HBA BIOS, see your vendor-specific documentation.

3. Reboot the host and verify that the OS is up and running.

Step 2: Install RHEL and NVMe software and verify your configuration

To configure your host for NVMe-oF you need to install the host and NVMe software packages, enable multipathing, and verify your host NQN configuration.

Steps

1. Install RHEL 8.x on the server. After the installation is complete, verify that you are running the required RHEL 8.x kernel:

```
uname -r
```

   Example RHEL kernel version:

```
4.18.0-553.el8_10.x86_64
```

2. Install the `nvme-cli` package:

```
rpm -qa|grep nvme-cli
```

   The following example shows an nvme-cli package version:

```
nvme-cli-1.16-9.el8.x86_64
```

3. Install the `libnvme` package:

```
rpm -qa|grep libnvme
```

   The following example shows an libnvme package version:

```
libnvme-1.4-3.el8.x86_64
```

4. Enable in-kernel NVMe multipath:

```
grubby --args=nvme_core.multipath=Y --update-kernel /boot/vmlinuz-
4.18.0-553.el8_10.x86_64
```

5. On the RHEL 8.x host, check the `hostnqn` string at `/etc/nvme/hostnqn`:

```
cat /etc/nvme/hostnqn
```

   The following example shows a `hostnqn` version:

```
nqn.2014-08.org.nvmexpress:uuid:4c4c4544-0032-3410-8035-b8c04f4c5132
```

6. On the ONTAP system, verify that the `hostnqn` string matches the `hostnqn` string for the corresponding subsystem on the ONTAP storage system:

```
::> vserver nvme subsystem host show -vserver vs_fcnvme_141
```

Show example

```
Vserver     Subsystem          Host NQN
----------- ---------------
------------------------------------------------------------
vs_25_2742   rhel_101_QLe2772    nqn.2014-
08.org.nvmexpress:uuid:546399fc-160f-11e5-89aa-98be942440ca
```

If the `hostnqn` strings do not match, use the `vserver modify` command to update the `hostnqn` string on your corresponding ONTAP storage system subsystem to match the `hostnqn` string from `/etc/nvme/hostnqn` on the host.

7. Reboot the host.

To run both NVMe and SCSI traffic on the same host, NetApp recommends using the in-kernel NVMe multipath for ONTAP namespaces and dm-multipath for ONTAP LUNs. To prevent dm-multipath from claiming ONTAP namespace devices, exclude them by adding the `enable_foreign` setting to the `/etc/multipath.conf` file:

```
cat /etc/multipath.conf
defaults {
        enable_foreign     NONE
}
```

Step 3: Configure NVMe/FC and NVMe/TCP

Configure NVMe/FC with Broadcom/Emulex or Marvell/QLogic adapters, or configure NVMe/TCP using manual discovery and connect operations.

NVMe/FC - Broadcom/Emulex

Configure NVMe/FC for a Broadcom/Emulex adapter.

Steps

1. Verify that you are using the supported adapter model:
   a. Display the model names:

```
cat /sys/class/scsi_host/host*/modelname
```

   You should see the following output:

```
LPe32002-M2
LPe32002-M2
```

   b. Display the model descriptions:

```
cat /sys/class/scsi_host/host*/modeldesc
```

   You should see an output similar to the following example:

```
Emulex LightPulse LPe32002-M2 2-Port 32Gb Fibre Channel Adapter
Emulex LightPulse LPe32002-M2 2-Port 32Gb Fibre Channel Adapter
```

2. Verify that you are using the recommended Broadcom `lpfc` firmware and inbox driver:
   a. Display the firmware version:

```
cat /sys/class/scsi_host/host*/fwrev
```

   The command returns the firmware versions:

```
14.2.539.21, sli-4:2:c
14.2.539.21, sli-4:2:c
```

   b. Display the inbox driver version:

```
cat /sys/module/lpfc/version
```

   The following example shows a driver version:

```
0:14.0.0.21
```

For the current list of supported adapter driver and firmware versions, see the Interoperability Matrix Tool.

3. Verify that `lpfc_enable_fc4_type` is set to `3`:

```
cat /sys/module/lpfc/parameters/lpfc_enable_fc4_type
```

4. Verify that you can view your initiator ports:

```
cat /sys/class/fc_host/host*/port_name
```

You should see an output similar to:

```
0x10000090fae0ec88
0x10000090fae0ec89
```

5. Verify that your initiator ports are online:

```
cat /sys/class/fc_host/host*/port_state
```

You should see the following output:

```
Online
Online
```

6. Verify that the NVMe/FC initiator ports are enabled and that the target ports are visible:

```
cat /sys/class/scsi_host/host*/nvme_info
```

Show example

```
NVME Initiator Enabled
XRI Dist lpfc0 Total 6144 IO 5894 ELS 250
NVME LPORT lpfc0 WWPN x100000109bf044b1 WWNN x200000109bf044b1
DID x022a00 ONLINE
NVME RPORT       WWPN x211ad039eaa7dfc8 WWNN x2119d039eaa7dfc8
DID x021302 TARGET DISCSRVC ONLINE
NVME RPORT       WWPN x211cd039eaa7dfc8 WWNN x2119d039eaa7dfc8
DID x020b02 TARGET DISCSRVC ONLINE

NVME Statistics
LS: Xmt 00000001ff Cmpl 00000001ff Abort 00000000
LS XMIT: Err 00000000  CMPL: xb 00000000 Err 00000000
Total FCP Cmpl 0000000001330ec7 Issue 0000000001330ec9 OutIO
0000000000000002
        abort 00000330 noxri 00000000 nondlp 0000000b qdepth
00000000 wqerr 00000000 err 00000000
FCP CMPL: xb 00000354 Err 00000361

NVME Initiator Enabled
XRI Dist lpfc1 Total 6144 IO 5894 ELS 250
NVME LPORT lpfc1 WWPN x100000109bf044b2 WWNN x200000109bf044b2
DID x021b00 ONLINE
NVME RPORT       WWPN x211bd039eaa7dfc8 WWNN x2119d039eaa7dfc8
DID x022902 TARGET DISCSRVC ONLINE
NVME RPORT       WWPN x211dd039eaa7dfc8 WWNN x2119d039eaa7dfc8
DID x020102 TARGET DISCSRVC ONLINE

NVME Statistics
LS: Xmt 00000001ff Cmpl 00000001ff Abort 00000000
LS XMIT: Err 00000000  CMPL: xb 00000000 Err 00000000
Total FCP Cmpl 00000000012ec220 Issue 00000000012ec222 OutIO
0000000000000002
        abort 0000033b noxri 00000000 nondlp 00000085 qdepth
00000000 wqerr 00000000 err 00000000
FCP CMPL: xb 00000368 Err 00000382
```

NVMe/FC - Marvell/QLogic

Configure NVMe/FC for a Marvell/QLogic adapter.

Steps

1. Verify that you are using the supported adapter driver and firmware versions:

```
cat /sys/class/fc_host/host*/symbolic_name
```

The following example shows driver and firmware versions:

```
QLE2742 FW: v9.10.11 DVR: v10.02.08.200-k
QLE2742 FW: v9.10.11 DVR: v10.02.08.200-k
```

2. Verify that `ql2xnvmeenable` is set. This enables the Marvell adapter to function as an NVMe/FC initiator:

```
cat /sys/module/qla2xxx/parameters/ql2xnvmeenable
```

The expected output is 1.

NVMe/TCP

The NVMe/TCP protocol doesn't support the auto-connect operation. Instead, you can discover the NVMe/TCP subsystems and namespaces by performing the NVMe/TCP `connect` or `connect-all` operations manually.

Steps

1. Check that the initiator port can get the discovery log page data across the supported NVMe/TCP LIFs:

```
nvme discover -t tcp -w host-traddr -a traddr
```

Show example

```
nvme discover -t tcp -w 192.168.2.31 -a 192.168.2.25

Discovery Log Number of Records 8, Generation counter 18
=====Discovery Log Entry 0======
trtype:  tcp
adrfam:  ipv4
subtype: unrecognized
treq:    not specified.
portid:  0
trsvcid: 8009
subnqn:  nqn.1992-
08.com.netapp:sn.a1b2b785b9de11ee8e7fd039ea9e8ae9:discovery
traddr:  192.168.1.25
sectype: none
=====Discovery Log Entry 1======
trtype:  tcp
adrfam:  ipv4
subtype: unrecognized
treq:    not specified.
portid:  1
trsvcid: 8009
subnqn:  nqn.1992-
08.com.netapp:sn.a1b2b785b9de11ee8e7fd039ea9e8ae9:discovery
traddr:  192.168.2.26
sectype: none
```

2. Verify that the other NVMe/TCP initiator-target LIF combinations can successfully fetch discovery log page data:

```
nvme discover -t tcp -w host-traddr -a traddr
```

Show example

```
nvme  discover  -t  tcp  -w 192.168.2.31  -a 192.168.2.25
nvme  discover  -t  tcp  -w 192.168.1.31  -a 192.168.1.24
nvme  discover  -t  tcp  -w 192.168.2.31  -a 192.168.2.26
nvme  discover  -t  tcp  -w 192.168.1.31  -a 192.168.1.25
```

3. Run the `nvme connect-all` command across all the supported NVMe/TCP initiator-target LIFs across the nodes:

```
nvme connect-all -t tcp -w host-traddr -a traddr
```

Show example

```
nvme connect-all -t tcp -w 192.168.2.31 -a 192.168.2.25 -l 1800
nvme connect-all -t tcp -w 192.168.1.31 -a 192.168.1.24 -l 1800
nvme connect-all -t tcp -w 192.168.2.31 -a 192.168.2.26 -l 1800
nvme connect-all -t tcp -w 192.168.1.31 -a 192.168.1.25 -l 1800
```

Step 4: Optionally, enable 1MB I/O for NVMe/FC

ONTAP reports a Max Data Transfer Size (MDTS) of 8 in the Identify Controller data. This means the maximum I/O request size can be up to 1MB. To issue I/O requests of size 1MB for a Broadcom NVMe/FC host, you should increase the `lpfc` value of the `lpfc_sg_seg_cnt` parameter to 256 from the default value of 64.

These steps don't apply to Qlogic NVMe/FC hosts.

Steps

1. Set the `lpfc_sg_seg_cnt` parameter to 256:

   ```
   cat /etc/modprobe.d/lpfc.conf
   ```

   You should see an output similar to the following example:

   ```
   options lpfc lpfc_sg_seg_cnt=256
   ```

2. Run the `dracut -f` command, and reboot the host.
3. Verify that the value for `lpfc_sg_seg_cnt` is 256:

   ```
   cat /sys/module/lpfc/parameters/lpfc_sg_seg_cnt
   ```

Step 5: Verify the multipathing configuration

Verify that the in-kernel NVMe multipath status, ANA status, and ONTAP namespaces are correct for the NVMe-oF configuration.

Steps

1. Verify that the in-kernel NVMe multipath is enabled:

```
cat /sys/module/nvme_core/parameters/multipath
```

You should see the following output:

```
Y
```

2. Verify that the appropriate NVMe-oF settings (such as, model set to NetApp ONTAP Controller and load balancing iopolicy set to round-robin) for the respective ONTAP namespaces correctly reflect on the host:
   a. Display the subsystems:

      ```
      cat /sys/class/nvme-subsystem/nvme-subsys*/model
      ```

      You should see the following output:

      ```
      NetApp ONTAP Controller
      NetApp ONTAP Controller
      ```

   b. Display the policy:

      ```
      cat /sys/class/nvme-subsystem/nvme-subsys*/iopolicy
      ```

      You should see the following output:

      ```
      round-robin
      round-robin
      ```

3. Verify that the namespaces are created and correctly discovered on the host:

```
nvme list
```

Show example

```
Node         SN                   Model
-------------------------------------------------------
/dev/nvme4n1 81Ix2BVuekWcAAAAAAAB   NetApp ONTAP Controller


Namespace Usage    Format              FW              Rev
-------------------------------------------------------------
1                 21.47 GB / 21.47 GB   4 KiB + 0 B   FFFFFFFF
```

4. Verify that the controller state of each path is live and has the correct ANA status:

```
nvme list-subsys /dev/nvme0n1
```

Show NVMe/FC example

```
nvme-subsys0 - NQN=nqn.1992-
08.com.netapp:sn.0cd9ee0dc0ec11ee8e7fd039ea9e8ae9:subsystem.nvme
\
 +- nvme1 fc traddr=nn-0x2005d039eaa7dfc8:pn-0x2086d039eaa7dfc8
host_traddr=nn-0x20000024ff752e6d:pn-0x21000024ff752e6d live non-
optimized
 +- nvme2 fc traddr=nn-0x2005d039eaa7dfc8:pn-0x2016d039eaa7dfc8
host_traddr=nn-0x20000024ff752e6c:pn-0x21000024ff752e6c live
optimized
 +- nvme3 fc traddr=nn-0x2005d039eaa7dfc8:pn-0x2081d039eaa7dfc8
host_traddr=nn-0x20000024ff752e6c:pn-0x21000024ff752e6c live non-
optimized
 +- nvme4 fc traddr=nn-0x2005d039eaa7dfc8:pn-0x2087d039eaa7dfc8
host_traddr=nn-0x20000024ff752e6d:pn-0x21000024ff752e6d live
optimized
```

Show NVMe/TCP example

```
nvme-subsys0 - NQN=nqn.1992-
08.com.netapp:sn.a1b2b785b9de11ee8e7fd039ea9e8ae9:subsystem.nvme_tcp
_1
\
 +- nvme0 tcp traddr=192.168.2.26 trsvcid=4420
host_traddr=192.168.2.31 live non-optimized
 +- nvme1 tcp traddr=192.168.2.25 trsvcid=4420
host_traddr=192.168.2.31 live optimized
 +- nvme2 tcp traddr=192.168.1.25 trsvcid=4420
host_traddr=192.168.1.31 live non-optimized
 +- nvme3 tcp traddr=192.168.1.24 trsvcid=4420
host_traddr=192.168.1.31 live optimized
```

5. Verify that the namespaces are created and correctly discovered on the host:

```
nvme list
```

Show example

```
Node         SN                   Model
------------------------------------------------------------
/dev/nvme4n1 81Ix2BVuekWcAAAAAAAB   NetApp ONTAP Controller


Namespace Usage    Format             FW             Rev
------------------------------------------------------------
1                 21.47 GB / 21.47 GB   4 KiB + 0 B   FFFFFFFF
```

6. Verify that the NetApp plug-in displays the correct values for each ONTAP namespace device:

Column

```
nvme netapp ontapdevices -o column
```

Show example

```
Device        Vserver       Namespace Path
-----------------------------------------------------
/dev/nvme0n1  tcpiscsi_129  /vol/tcpnvme_1_0_0/tcpnvme_ns



NSID       UUID                                   Size
------------------------------------------------------------
1          05c2c351-5d7f-41d7-9bd8-1a56c            21.47GB
```

JSON

```
nvme netapp ontapdevices -o json
```

Show example

```
{
  "ONTAPdevices": [
    {
      "Device": "/dev/nvme0n1",
      "Vserver": "tcpiscsi_129",
      "Namespace Path”: /vol/tcpnvme_1_0_0/tcpnvme_ns ",
      "NSID": 1,
      "UUID": " 05c2c351-5d7f-41d7-9bd8-1a56c160c80b ",
      "Size2: "21.47GB",
      "LBA_Data_Size": 4096,
      "Namespace Size" : 5242880
    },
  ]
}
```

Step 6: Review the known issues

These are the known issues:

NetApp Bug ID	Title	Description
1479047	RHEL 8.x NVMe-oF hosts create duplicate persistent discovery controllers (PDCs)	On NVMe-oF hosts, you can use the "nvme discover -p" command to create PDCs. When this command is used, only one PDC should be created per initiator-target combination. However, if you are running RHEL 8.x on an NVMe-oF host, a duplicate PDC is created each time "nvme discover -p" is executed. This leads to unnecessary usage of resources on both the host and the target.

Rocky Linux

Learn about ONTAP support and features for Rocky Linux

Features supported for host configuration with NVMe over Fabrics (NVMe-oF) varies based on your version of ONTAP and Rocky Linux.

Feature	Rocky Linux host version	ONTAP version
Secure in-band authentication is supported over NVMe/TCP between a RHEL host and an ONTAP controller	9.3 or later	9.12.1 or later
NVMe/TCP provides namespaces using the native `nvme-cli` package	8.2 or later	9.10.1 or later
NVMe/TCP is a fully supported enterprise feature	9.0 or later	9.10.1 or later
NVMe and SCSI traffic is supported on the same host using NVMe multipath for NVMe-oF namespaces and dm-multipath for SCSI LUNs	8.2 or later	9.4 or later

ONTAP supports the following SAN host features regardless of the ONTAP version running on your system setup.

Feature	Rocky Linux host version
Native NVMe multipathing is enabled by default	10.0 or later
SAN booting is enabled using the NVMe/FC protocol	9.4 or later
The `nvme-cli` package includes auto-connect scripts removing the need for third-party scripts	8.2 or later
The native udev rule in the `nvme-cli` package provides round-robin load balancing for NVMe multipathing	8.2 or later

For details on supported configurations, see the Interoperability Matrix Tool.

What's next

If your Rocky Linux version is ..	Learn about ..
10 series	Configuring NVMe for Rocky Linux 10.x
9 series	Configuring NVMe for Rocky Linux 9.x
8 series	Configuring NVMe for Rocky Linux 8.x

Related information

Learn about managing NVMe protocols

Configure Rocky Linux 10.x for NVMe-oF with ONTAP storage

Rocky Linux hosts support the NVMe over Fibre Channel (NVMe/FC) and NVMe over TCP (NVMe/TCP) protocols with Asymmetric Namespace Access (ANA). ANA provides multipathing functionality equivalent to asymmetric logical unit access (ALUA) in iSCSI and FCP environments.

Learn how to configure NVMe over Fabrics (NVMe-oF) hosts for Rocky Linux 10.x. For more support and feature information, see Rocky Linux ONTAP support and features.

NVMe-oF with Rocky Linux 10.x has the following known limitations:

- The `nvme disconnect-all` command disconnects both root and data filesystems and might lead to system instability. Do not issue this on systems booting from SAN over NVMe-TCP or NVMe-FC namespaces.

Step 1: Optionally, enable SAN booting

You can configure your host to use SAN booting to simplify deployment and improve scalability. Use the Interoperability Matrix Tool to verify that your Linux OS, host bus adapter (HBA), HBA firmware, HBA boot BIOS, and ONTAP version support SAN booting.

Steps

1. Create a NVMe namespace and map it to the host.
2. Enable SAN booting in the server BIOS for the ports to which the SAN boot namespace is mapped.

   For information on how to enable the HBA BIOS, see your vendor-specific documentation.
3. Reboot the host and verify that the OS is up and running.

Step 2: Install Rocky Linux and NVMe software and verify your configuration

To configure your host for NVMe-oF you need to install the host and NVMe software packages, enable multipathing, and verify your host NQN configuration.

Steps

1. Install Rocky Linux 10.x on the server. After the installation is complete, verify that you are running the

required Rocky Linux 10.x kernel:

```
uname -r
```

Example Rocky Linux kernel version:

```
6.12.0-55.9.1.el10_0.x86_64
```

2. Install the `nvme-cli` package:

```
rpm -qa|grep nvme-cli
```

The following example shows an `nvme-cli` package version:

```
nvme-cli-2.11-5.el10.x86_64
```

3. Install the `libnvme` package:

```
rpm -qa|grep libnvme
```

The following example shows an `libnvme` package version:

```
libnvme-1.11.1-1.el10.x86_64
```

4. On the host, check the hostnqn string at `/etc/nvme/hostnqn`:

```
cat /etc/nvme/hostnqn
```

The following example shows a `hostnqn` value:

```
nqn.2014-08.org.nvmexpress:uuid:4c4c4544-0056-5410-8048-b9c04f425633
```

5. On the ONTAP system, verify that the `hostnqn` string matches the `hostnqn` string for the corresponding subsystem on the ONTAP array:

```
::> vserver nvme subsystem host show -vserver vs_nvme_194_rockylinux10
```

Show example

```
Vserver Subsystem Priority  Host NQN
------- --------- --------
-------------------------------------------------
vs_nvme_194_rockylinux10
        nvme4
                  regular   nqn.2014-
08.org.nvmexpress:uuid:4c4c4544-0056-5410-8048-c7c04f425633
        nvme_1
                  regular   nqn.2014-
08.org.nvmexpress:uuid:4c4c4544-0056-5410-8048-c7c04f425633
        nvme_2
                  regular   nqn.2014-
08.org.nvmexpress:uuid:4c4c4544-0056-5410-8048-c7c04f425633
        nvme_3
                  regular   nqn.2014-
08.org.nvmexpress:uuid:4c4c4544-0056-5410-8048-c7c04f425633
4 entries were displayed.
```

If the `hostnqn` strings do not match, use the `vserver modify` command to update the `hostnqn` string on your corresponding ONTAP storage system subsystem to match the `hostnqn` string from `/etc/nvme/hostnqn` on the host.

Step 3: Configure NVMe/FC and NVMe/TCP

Configure NVMe/FC with Broadcom/Emulex or Marvell/QLogic adapters, or configure NVMe/TCP using manual discovery and connect operations.

NVMe/FC - Broadcom/Emulex

Configure NVMe/FC for a Broadcom/Emulex adapter.

Steps

1. Verify that you are using the supported adapter model:
   a. Display the model names:

```
cat /sys/class/scsi_host/host*/modelname
```

   You should see the following output:

```
LPe36002-M64
LPe36002-M64
```

   b. Display the model descriptions:

```
cat /sys/class/scsi_host/host*/modeldesc
```

   You should see an output similar to the following example:

```
Emulex LightPulse LPe36002-M64 2-Port 64Gb Fibre Channel Adapter
Emulex LightPulse LPe36002-M64 2-Port 64Gb Fibre Channel Adapter
```

2. Verify that you are using the recommended Broadcom `lpfc` firmware and inbox driver:
   a. Display the firmware version:

```
cat /sys/class/scsi_host/host*/fwrev
```

   The command returns the firmware versions:

```
14.0.539.16, sli-4:6:d
14.0.539.16, sli-4:6:d
```

   b. Display the inbox driver version:

```
cat /sys/module/lpfc/version
```

   The following example shows a driver version:

```
0:14.4.0.6
```

For the current list of supported adapter driver and firmware versions, see the Interoperability Matrix Tool.

3. Verify that `lpfc_enable_fc4_type` is set to `3`:

```
cat /sys/module/lpfc/parameters/lpfc_enable_fc4_type
```

4. Verify that you can view your initiator ports:

```
cat /sys/class/fc_host/host*/port_name
```

You should see an output similar to:

```
0x2100f4c7aa0cd7c2
0x2100f4c7aa0cd7c3
```

5. Verify that your initiator ports are online:

```
cat /sys/class/fc_host/host*/port_state
```

You should see the following output:

```
Online
Online
```

6. Verify that the NVMe/FC initiator ports are enabled and that the target ports are visible:

```
cat /sys/class/scsi_host/host*/nvme_info
```

Show example

```
NVME Initiator Enabled
XRI Dist lpfc2 Total 6144 IO 5894 ELS 250
NVME LPORT lpfc2 WWPN x100000109bf044b1 WWNN x200000109bf044b1
DID x022a00 ONLINE
NVME RPORT       WWPN x202fd039eaa7dfc8 WWNN x202cd039eaa7dfc8
DID x021310 TARGET DISCSRVC ONLINE
NVME RPORT       WWPN x202dd039eaa7dfc8 WWNN x202cd039eaa7dfc8
DID x020b10 TARGET DISCSRVC ONLINE

NVME Statistics
LS: Xmt 0000000810 Cmpl 0000000810 Abort 00000000
LS XMIT: Err 00000000  CMPL: xb 00000000 Err 00000000
Total FCP Cmpl 000000007b098f07 Issue 000000007aee27c4 OutIO
fffffffffe498bd
        abort 000013b4 noxri 00000000 nondlp 00000058 qdepth
00000000 wqerr 00000000 err 00000000
FCP CMPL: xb 000013b4 Err 00021443

NVME Initiator Enabled
XRI Dist lpfc3 Total 6144 IO 5894 ELS 250
NVME LPORT lpfc3 WWPN x100000109bf044b2 WWNN x200000109bf044b2
DID x021b00 ONLINE
NVME RPORT       WWPN x2033d039eaa7dfc8 WWNN x202cd039eaa7dfc8
DID x020110 TARGET DISCSRVC ONLINE
NVME RPORT       WWPN x2032d039eaa7dfc8 WWNN x202cd039eaa7dfc8
DID x022910 TARGET DISCSRVC ONLINE

NVME Statistics
LS: Xmt 0000000840 Cmpl 0000000840 Abort 00000000
LS XMIT: Err 00000000  CMPL: xb 00000000 Err 00000000
Total FCP Cmpl 000000007afd4434 Issue 000000007ae31b83 OutIO
fffffffffe5d74f
        abort 000014a5 noxri 00000000 nondlp 0000006a qdepth
00000000 wqerr 00000000 err 00000000
FCP CMPL: xb 000014a5 Err 0002149a
```

NVMe/FC - Marvell/QLogic

Configure NVMe/FC for a Marvell/QLogic adapter.

Steps

1. Verify that you are using the supported adapter driver and firmware versions:

```
cat /sys/class/fc_host/host*/symbolic_name
```

The following example shows driver and firmware versions:

```
QLE2872 FW:v9.15.00 DVR:v10.02.09.300-k
QLE2872 FW:v9.15.00 DVR:v10.02.09.300-k
```

2. Verify that `ql2xnvmeenable` is set. This enables the Marvell adapter to function as an NVMe/FC initiator:

```
cat /sys/module/qla2xxx/parameters/ql2xnvmeenable
```

The expected output is 1.

NVMe/TCP

The NVMe/TCP protocol doesn't support the auto-connect operation. Instead, you can discover the NVMe/TCP subsystems and namespaces by performing the NVMe/TCP `connect` or `connect-all` operations manually.

Steps

1. Check that the initiator port can get the discovery log page data across the supported NVMe/TCP LIFs:

```
nvme discover -t tcp -w host-traddr -a traddr
```

Show example

```
nvme discover -t tcp -w 192.168.20.1 -a 192.168.20.20

Discovery Log Number of Records 8, Generation counter 18
=====Discovery Log Entry 0======
trtype:  tcp
adrfam:  ipv4
subtype: current discovery subsystem
treq:    not specified
portid:  4
trsvcid: 8009
subnqn:  nqn.1992-
08.com.netapp:sn.64e65e6caae711ef9668d039ea951c46:discovery
traddr:  192.168.21.21
eflags:  explicit discovery connections, duplicate discovery
information
sectype: none
=====Discovery Log Entry 1======
trtype:  tcp
adrfam:  ipv4
subtype: current discovery subsystem
treq:    not specified
portid:  2
trsvcid: 8009
subnqn:  nqn.1992-
08.com.netapp:sn.64e65e6caae711ef9668d039ea951c46:discovery
traddr:  192.168.20.21
eflags:  explicit discovery connections, duplicate discovery
information
sectype: none
=====Discovery Log Entry 2======
trtype:  tcp
adrfam:  ipv4
subtype: current discovery subsystem
treq:    not specified
portid:  3
trsvcid: 8009
subnqn:  nqn.1992-
08.com.netapp:sn.64e65e6caae711ef9668d039ea951c46:discovery
traddr:  192.168.21.20
eflags:  explicit discovery connections, duplicate discovery
information
sectype: none
=====Discovery Log Entry 3======
trtype:  tcp
```

```
adrfam:  ipv4
subtype: current discovery subsystem
treq:    not specified
portid:  1
trsvcid: 8009
subnqn:  nqn.1992-
08.com.netapp:sn.64e65e6caae711ef9668d039ea951c46:discovery
traddr:  192.168.20.20
eflags:  explicit discovery connections, duplicate discovery
information
sectype: none
=====Discovery Log Entry 4======
trtype:  tcp
adrfam:  ipv4
subtype: nvme subsystem
treq:    not specified
portid:  4
trsvcid: 4420
subnqn:  nqn.1992-
08.com.netapp:sn.64e65e6caae711ef9668d039ea951c46:subsystem.rock
ylinux10_tcp_subsystem
traddr:  192.168.21.21
eflags:  none
sectype: none
=====Discovery Log Entry 5======
trtype:  tcp
adrfam:  ipv4
subtype: nvme subsystem
treq:    not specified
portid:  2
trsvcid: 4420
subnqn:  nqn.1992-
08.com.netapp:sn.64e65e6caae711ef9668d039ea951c46:subsystem.rock
ylinux10_tcp_subsystem
traddr:  192.168.20.21
eflags:  none
sectype: none
=====Discovery Log Entry 6======
trtype:  tcp
adrfam:  ipv4
subtype: nvme subsystem
treq:    not specified
portid:  3
trsvcid: 4420
subnqn:  nqn.1992-
08.com.netapp:sn.64e65e6caae711ef9668d039ea951c46:subsystem.rock
```

```
ylinux10_tcp_subsystem
traddr:  192.168.21.20
eflags:  none
sectype: none
=====Discovery Log Entry 7======
trtype:  tcp
adrfam:  ipv4
subtype: nvme subsystem
treq:    not specified
portid:  1
trsvcid: 4420
subnqn:  nqn.1992-
08.com.netapp:sn.64e65e6caae711ef9668d039ea951c46:subsystem.rock
ylinux10_tcp_subsystem
traddr:  192.168.20.20
eflags:  none
sectype: none
```

2. Verify that the other NVMe/TCP initiator-target LIF combinations can successfully retrieve discovery log page data:

```
nvme discover -t tcp -w host-traddr -a traddr
```

Show example

```
nvme discover -t tcp -w 192.168.20.1 -a 192.168.20.20
nvme discover -t tcp -w 192.168.21.1 -a 192.168.21.20
nvme discover -t tcp -w 192.168.20.1 -a 192.168.20.21
nvme discover -t tcp -w 192.168.21.1 -a 192.168.21.21
```

3. Run the `nvme connect-all` command across all the supported NVMe/TCP initiator-target LIFs across the nodes:

```
nvme connect-all -t tcp -w host-traddr -a traddr
```

Show example

```
nvme    connect-all -t  tcp -w  192.168.20.1    -a
192.168.20.20
nvme    connect-all -t  tcp -w  192.168.21.1    -a
192.168.21.20
nvme    connect-all -t  tcp -w  192.168.20.1    -a
192.168.20.21
nvme    connect-all -t  tcp -w  192.168.21.1    -a
192.168.21.21
```

Beginning with Rocky Linux 9.4, the setting for the NVMe/TCP `ctrl_loss_tmo timeout` is automatically set to "off". As a result:

- There are no limits on the number of retries (indefinite retry).
- You don't need to manually configure a specific `ctrl_loss_tmo timeout` duration when using the `nvme connect` or `nvme connect-all` commands (option -l).
- The NVMe/TCP controllers don't experience timeouts in the event of a path failure and remain connected indefinitely.

Step 4: Optionally, enable 1MB I/O for NVMe/FC

ONTAP reports a Max Data Transfer Size (MDTS) of 8 in the Identify Controller data. This means the maximum I/O request size can be up to 1MB. To issue I/O requests of size 1MB for a Broadcom NVMe/FC host, you should increase the `lpfc` value of the `lpfc_sg_seg_cnt` parameter to 256 from the default value of 64.

These steps don't apply to Qlogic NVMe/FC hosts.

Steps

1. Set the `lpfc_sg_seg_cnt` parameter to 256:

   ```
   cat /etc/modprobe.d/lpfc.conf
   ```

   You should see an output similar to the following example:

   ```
   options lpfc lpfc_sg_seg_cnt=256
   ```

2. Run the `dracut -f` command, and reboot the host.
3. Verify that the value for `lpfc_sg_seg_cnt` is 256:

```
cat /sys/module/lpfc/parameters/lpfc_sg_seg_cnt
```

Step 5: Verify NVMe boot services

The `nvmefc-boot-connections.service` and `nvmf-autoconnect.service` boot services included in the NVMe/FC `nvme-cli` package are automatically enabled when the system boots.

After booting completes, verify that the `nvmefc-boot-connections.service` and `nvmf-autoconnect.service` boot services are enabled.

Steps

1. Verify that `nvmf-autoconnect.service` is enabled:

```
systemctl status nvmf-autoconnect.service
```

Show example output

```
nvmf-autoconnect.service - Connect NVMe-oF subsystems automatically
during boot
     Loaded: loaded (/usr/lib/systemd/system/nvmf-
autoconnect.service; enabled; preset: disabled)
     Active: inactive (dead)

Jun 10 04:06:26 SR630-13-201.lab.eng.btc.netapp.in systemd[1]:
Starting Connect NVMe-oF subsystems automatically during boot...
Jun 10 04:06:26 SR630-13-201.lab.eng.btc.netapp.in systemd[1]: nvmf-
autoconnect.service: Deactivated successfully.
Jun 10 04:06:26 SR630-13-201.lab.eng.btc.netapp.in systemd[1]:
Finished Connect NVMe-oF subsystems automatically during boot.
```

2. Verify that `nvmefc-boot-connections.service` is enabled:

```
systemctl status nvmefc-boot-connections.service
```

Show example output

```
nvmefc-boot-connections.service - Auto-connect to subsystems on FC-
NVME devices found during boot
     Loaded: loaded (/usr/lib/systemd/system/nvmefc-boot-
connections.service; enabled; preset: enabled)
     Active: inactive (dead) since Tue 2025-06-10 01:08:36 EDT; 2h
59min ago
   Main PID: 7090 (code=exited, status=0/SUCCESS)
        CPU: 30ms

Jun 10 01:08:36 localhost systemd[1]: Starting Auto-connect to
subsystems on FC-NVME devices found during boot...
Jun 10 01:08:36 localhost systemd[1]: nvmefc-boot-
connections.service: Deactivated successfully.
Jun 10 01:08:36 localhost systemd[1]: Finished Auto-connect to
subsystems on FC-NVME devices found during boot.
```

Step 6: Verify the multipathing configuration

Verify that the in-kernel NVMe multipath status, ANA status, and ONTAP namespaces are correct for the NVMe-oF configuration.

Steps

1. Verify that the in-kernel NVMe multipath is enabled:

   ```
   cat /sys/module/nvme_core/parameters/multipath
   ```

   You should see the following output:

   ```
   Y
   ```

2. Verify that the appropriate NVMe-oF settings (such as, model set to NetApp ONTAP Controller and load balancing iopolicy set to round-robin) for the respective ONTAP namespaces correctly reflect on the host:

   a. Display the subsystems:

      ```
      cat /sys/class/nvme-subsystem/nvme-subsys*/model
      ```

      You should see the following output:

```
NetApp ONTAP Controller
NetApp ONTAP Controller
```

b. Display the policy:

```
cat /sys/class/nvme-subsystem/nvme-subsys*/iopolicy
```

You should see the following output:

```
round-robin
round-robin
```

3. Verify that the namespaces are created and correctly discovered on the host:

```
nvme list
```

Show example

```
Node         SN                   Model
------------------------------------------------------------
/dev/nvme4n1 81Ix2BVuekWcAAAAAAAB   NetApp ONTAP Controller


Namespace Usage    Format             FW             Rev
------------------------------------------------------------
1                 21.47 GB / 21.47 GB   4 KiB + 0 B   FFFFFFFF
```

4. Verify that the controller state of each path is live and has the correct ANA status:

NVMe/FC

```
nvme list-subsys /dev/nvme5n1
```

Show example

```
nvme-subsys5 - NQN=nqn.1992-
08.com.netapp:sn.f7565b15a66911ef9668d039ea951c46:subsystem.nvme
1
               hostnqn=nqn.2014-08.org.nvmexpress:uuid:4c4c4544-
0056-5410-8048-c7c04f425633
\
 +- nvme126 fc traddr=nn-0x2036d039ea951c45:pn-
0x2038d039ea951c45,host_traddr=nn-0x2000f4c7aa0cd7c3:pn-
0x2100f4c7aa0cd7c3 live optimized
 +- nvme176 fc traddr=nn-0x2036d039ea951c45:pn-
0x2037d039ea951c45,host_traddr=nn-0x2000f4c7aa0cd7c2:pn-
0x2100f4c7aa0cd7c2 live optimized
 +- nvme5 fc traddr=nn-0x2036d039ea951c45:pn-
0x2039d039ea951c45,host_traddr=nn-0x2000f4c7aa0cd7c2:pn-
0x2100f4c7aa0cd7c2 live non-optimized
 +- nvme71 fc traddr=nn-0x2036d039ea951c45:pn-
0x203ad039ea951c45,host_traddr=nn-0x2000f4c7aa0cd7c3:pn-
0x2100f4c7aa0cd7c3 live non-optimized
```

NVMe/TCP

```
nvme list-subsys /dev/nvme4n2
```

Show example

```
nvme-subsys4 - NQN=nqn.1992-
08.com.netapp:sn.64e65e6caae711ef9668d039ea951c46:subsystem.nvme
4
               hostnqn=nqn.2014-08.org.nvmexpress:uuid:4c4c4544-
0035-5910-804b-c2c04f444d33
\
+- nvme102 tcp
traddr=192.168.21.20,trsvcid=4420,host_traddr=192.168.21.1,src_a
ddr=192.168.21.1 live non-optimized
+- nvme151 tcp
traddr=192.168.21.21,trsvcid=4420,host_traddr=192.168.21.1,src_a
ddr=192.168.21.1 live optimized
+- nvme4 tcp
traddr=192.168.20.20,trsvcid=4420,host_traddr=192.168.20.1,src_a
ddr=192.168.20.1 live non-optimized
+- nvme53 tcp
traddr=192.168.20.21,trsvcid=4420,host_traddr=192.168.20.1,src_a
ddr=192.168.20.1 live optimized
```

5. Verify that the NetApp plug-in displays the correct values for each ONTAP namespace device:

Column

```
nvme netapp ontapdevices -o column
```

Show example

```
Device        Vserver   Namespace Path
---------------------- ------------------------------------
/dev/nvme10n1     vs_tcp_rockylinux10       /vol/vol10/ns10

NSID       UUID                                   Size
---------------------- ------------------------------------
1    bbf51146-fc64-4197-b8cf-8a24f6f359b3   21.47GB
```

JSON

```
nvme netapp ontapdevices -o json
```

Show example

```
{
  "ONTAPdevices":[
    {
      "Device":"/dev/nvme10n1",
      "Vserver":"vs_tcp_rockylinux10",
      "Namespace_Path":"/vol/vol10/ns10",
      "NSID":1,
      "UUID":"bbf51146-fc64-4197-b8cf-8a24f6f359b3",
      "Size":"21.47GB",
      "LBA_Data_Size":4096,
      "Namespace_Size":5242880
}
]
    }
```

Step 7: Set up secure in-band authentication

Secure in-band authentication is supported over NVMe/TCP between a Rocky Linux 10.x host and an ONTAP controller.

Each host or controller must be associated with a `DH-HMAC-CHAP` key to set up secure authentication. A `DH-HMAC-CHAP` key is a combination of the NQN of the NVMe host or controller and an authentication secret configured by the administrator. To authenticate its peer, an NVMe host or controller must recognize the key associated with the peer.

Steps

Set up secure in-band authentication using the CLI or a config JSON file. If you need to specify different dhchap keys for different subsystems, you must use a config JSON file.

CLI

Set up secure in-band authentication using the CLI.

1. Obtain the host NQN:

```
cat /etc/nvme/hostnqn
```

2. Generate the dhchap key for the Rocky Linux 10.x host.

   The following output describes the `gen-dhchap-key` command parameters:

```
nvme gen-dhchap-key -s optional_secret -l key_length {32|48|64} -m
HMAC_function {0|1|2|3} -n host_nqn
•   -s secret key in hexadecimal characters to be used to initialize
the host key
•   -l length of the resulting key in bytes
•   -m HMAC function to use for key transformation
0 = none, 1- SHA-256, 2 = SHA-384, 3=SHA-512
•   -n host NQN to use for key transformation
```

   In the following example, a random dhchap key with HMAC set to 3 (SHA-512) is generated.

```
nvme gen-dhchap-key -m 3 -n nqn.2014-
08.org.nvmexpress:uuid:4c4c4544-0035-5910-804b-c2c04f444d33
DHHC-
1:03:7zf8I9gaRcDWH3tCH5vLGaoyjzPIvwNWusBfKdpJa+hia1aKDKJQ2o53pX3wYM9
xdv5DtKNNhJInZ7X8wU2RQpQIngc=:
```

3. On the ONTAP controller, add the host and specify both dhchap keys:

```
vserver nvme subsystem host add -vserver <svm_name> -subsystem
<subsystem> -host-nqn <host_nqn> -dhchap-host-secret
<authentication_host_secret> -dhchap-controller-secret
<authentication_controller_secret> -dhchap-hash-function {sha-
256|sha-512} -dhchap-group {none|2048-bit|3072-bit|4096-bit|6144-
bit|8192-bit}
```

4. A host supports two types of authentication methods, unidirectional and bidirectional. On the host, connect to the ONTAP controller and specify dhchap keys based on the chosen authentication method:

```
nvme connect -t tcp -w <host-traddr> -a <tr-addr> -n <host_nqn> -S
<authentication_host_secret> -C <authentication_controller_secret>
```

5. Validate the `nvme connect authentication` command by verifying the host and controller dhchap keys:
   a. Verify the host dhchap keys:

```
cat /sys/class/nvme-subsystem/<nvme-subsysX>/nvme*/dhchap_secret
```

Show example output for a unidirectional configuration

```
cat /sys/class/nvme-subsystem/nvme-subsys1/nvme*/dhchap_secret
DHHC-
1:03:fMCrJharXUOqRoIsOEaG6m2PH1yYvu5+z3jTmzEKUbcWu26I33b93b
il2WR09XDho/ld3L45J+0FeCsStBEAfhYgkQU=:
DHHC-
1:03:fMCrJharXUOqRoIsOEaG6m2PH1yYvu5+z3jTmzEKUbcWu26I33b93b
il2WR09XDho/ld3L45J+0FeCsStBEAfhYgkQU=:
DHHC-
1:03:fMCrJharXUOqRoIsOEaG6m2PH1yYvu5+z3jTmzEKUbcWu26I33b93b
il2WR09XDho/ld3L45J+0FeCsStBEAfhYgkQU=:
DHHC-
1:03:fMCrJharXUOqRoIsOEaG6m2PH1yYvu5+z3jTmzEKUbcWu26I33b93b
il2WR09XDho/ld3L45J+0FeCsStBEAfhYgkQU=:
```

   b. Verify the controller dhchap keys:

```
cat /sys/class/nvme-subsystem/<nvme-
subsysX>/nvme*/dhchap_ctrl_secret
```

Show example output for a bidirectional configuration

```
cat /sys/class/nvme-subsystem/nvme-
subsys6/nvme*/dhchap_ctrl_secret
DHHC- 1:03:7zf8I9gaRcDWH3tCH5vLGaoyjzPIvwNWusBfKdpJa+hia
1aKDKJQ2o53pX3wYM9xdv5DtKNNhJInZ7X8wU2RQpQIngc=:

DHHC- 1:03:7zf8I9gaRcDWH3tCH5vLGaoyjzPIvwNWusBfKdpJa+hia
1aKDKJQ2o53pX3wYM9xdv5DtKNNhJInZ7X8wU2RQpQIngc=:

DHHC- 1:03:7zf8I9gaRcDWH3tCH5vLGaoyjzPIvwNWusBfKdpJa+hia
1aKDKJQ2o53pX3wYM9xdv5DtKNNhJInZ7X8wU2RQpQIngc=:

DHHC- 1:03:7zf8I9gaRcDWH3tCH5vLGaoyjzPIvwNWusBfKdpJa+hia
1aKDKJQ2o53pX3wYM9xdv5DtKNNhJInZ7X8wU2RQpQIngc=:
```

JSON

When multiple NVMe subsystems are available on the ONTAP controller, you can use the `/etc/nvme/config.json` file with the `nvme connect-all` command.

Use the `-o` option to generate the JSON file. Refer to the NVMe connect-all man pages for more syntax options.

1. Configure the JSON file.

> In the following example, `dhchap_key` corresponds to `dhchap_secret` and `dhchap_ctrl_key` corresponds to `dhchap_ctrl_secret`.

Show example

```
cat /etc/nvme/config.json
[
  {
    "hostnqn":"nqn.2014-08.org.nvmexpress:uuid:4c4c4544-0035-
5910-804b-c2c04f444d33",
    "hostid":"4c4c4544-0035-5910-804b-c2c04f444d33",
    "dhchap_key":"DHHC-
1:03:7zf8I9gaRcDWH3tCH5vLGaoyjzPIvwNWusBfKdpJa+hia1aKDKJQ2o53pX3
wYM9xdv5DtKNNhJInZ7X8wU2RQpQIngc=:",
    "subsystems":[
      {
        "nqn":"nqn.1992-
08.com.netapp:sn.127ade26168811f0a50ed039eab69ad3:subsystem.inba
nd_unidirectional",
        "ports":[
          {
            "transport":"tcp",
            "traddr":"192.168.20.17",
            "host_traddr":"192.168.20.1",
            "trsvcid":"4420"
          },
          {
            "transport":"tcp",
            "traddr":"192.168.20.18",
            "host_traddr":"192.168.20.1",
            "trsvcid":"4420"
          },
          {
            "transport":"tcp",
            "traddr":"192.168.21.18",
            "host_traddr":"192.168.21.1",
            "trsvcid":"4420"
          },
          {
            "transport":"tcp",
            "traddr":"192.168.21.17",
            "host_traddr":"192.168.21.1",
            "trsvcid":"4420"
}
]
```

2. Connect to the ONTAP controller using the config JSON file:

```
nvme connect-all -J /etc/nvme/config.json
```

Show example

```
traddr=192.168.20.20 is already connected
traddr=192.168.20.20 is already connected
traddr=192.168.20.20 is already connected
traddr=192.168.20.20 is already connected
traddr=192.168.20.20 is already connected
traddr=192.168.20.20 is already connected
traddr=192.168.20.20 is already connected
traddr=192.168.20.20 is already connected
traddr=192.168.20.21 is already connected
traddr=192.168.20.21 is already connected
traddr=192.168.20.21 is already connected
traddr=192.168.20.21 is already connected
traddr=192.168.20.21 is already connected
traddr=192.168.20.21 is already connected
traddr=192.168.20.21 is already connected
traddr=192.168.20.21 is already connected
```

3. Verify that the dhchap secrets have been enabled for the respective controllers for each subsystem.
   a. Verify the host dhchap keys:

      ```
      cat /sys/class/nvme-subsystem/nvme-subsys0/nvme0/dhchap_secret
      ```

      The following example shows a dhchap key:

      ```
      DHHC-1:03:7zf8I9gaRcDWH3tCH5vLGaoyjzPIvwNWusBfKdpJa+hia1
      aKDKJQ2o53pX3wYM9xdv5DtKNNhJInZ7X8wU2RQpQIngc=:
      ```

   b. Verify the controller dhchap keys:

      ```
      cat /sys/class/nvme-subsystem/nvme-
      subsys0/nvme0/dhchap_ctrl_secret
      ```

      You should see an output similar to the following example:

```
DHHC-1:03:fMCrJharXUOqRoIsOEaG6m2PH1yYvu5+z3jT
mzEKUbcWu26I33b93bil2WR09XDho/ld3L45J+0FeCsStBEAfhYgkQU=:
```

Step 8: Review the known issues

There are no known issues.

Configure Rocky Linux 9.x for NVMe-oF with ONTAP storage

Rocky Linux hosts support the NVMe over Fibre Channel (NVMe/FC) and NVMe over TCP (NVMe/TCP) protocols with Asymmetric Namespace Access (ANA). ANA provides multipathing functionality equivalent to asymmetric logical unit access (ALUA) in iSCSI and FCP environments.

Learn how to configure NVMe over Fabrics (NVMe-oF) hosts for Rocky Linux 9.x. For more support and feature information, see Rocky Linux ONTAP support and features.

NVMe-oF with Rocky Linux 9.x has the following known limitations:

- The `nvme disconnect-all` command disconnects both root and data filesystems and might lead to system instability. Do not issue this on systems booting from SAN over NVMe-TCP or NVMe-FC namespaces.

Step 1: Optionally, enable SAN booting

You can configure your host to use SAN booting to simplify deployment and improve scalability. Use the Interoperability Matrix Tool to verify that your Linux OS, host bus adapter (HBA), HBA firmware, HBA boot BIOS, and ONTAP version support SAN booting.

Steps

1. Create a NVMe namespace and map it to the host.
2. Enable SAN booting in the server BIOS for the ports to which the SAN boot namespace is mapped.

   For information on how to enable the HBA BIOS, see your vendor-specific documentation.

3. Reboot the host and verify that the OS is up and running.

Step 2: Install Rocky Linux and NVMe software and verify your configuration

To configure your host for NVMe-oF you need to install the host and NVMe software packages, enable multipathing, and verify your host NQN configuration.

Steps

1. Install Rocky Linux 9.x on the server. After the installation is complete, verify that you are running the required Rocky Linux 9.x kernel:

```
uname -r
```

Example Rocky Linux kernel version:

```
5.14.0-570.12.1.el9_6.x86_64
```

2. Install the `nvme-cli` package:

```
rpm -qa|grep nvme-cli
```

The following example shows an nvme-cli package version:

```
nvme-cli-2.11-5.el9.x86_64
```

3. Install the `libnvme` package:

```
rpm -qa|grep libnvme
```

The following example shows an `libnvme` package version:

```
libnvme-1.11.1-1.el9.x86_64
```

4. On the Rocky Linux host, check the hostnqn string at `/etc/nvme/hostnqn`:

```
cat /etc/nvme/hostnqn
```

The following example shows an `hostnqn` version:

```
nqn.2014-08.org.nvmexpress:uuid:4c4c4544-0056-5410-8048-b9c04f425633
```

5. On the ONTAP system, verify that the `hostnqn` string matches the `hostnqn` string for the corresponding subsystem on the ONTAP array:

```
::> vserver nvme subsystem host show -vserver vs_coexistence_LPE36002
```

Show example

```
Vserver Subsystem Priority  Host NQN
------- --------- --------
-------------------------------------------------
vs_coexistence_LPE36002
        nvme
                  regular   nqn.2014-
08.org.nvmexpress:uuid:4c4c4544-0056-5410-8048-b9c04f425633
        nvme_1
                  regular   nqn.2014-
08.org.nvmexpress:uuid:4c4c4544-0056-5410-8048-b9c04f425633
        nvme_2
                  regular   nqn.2014-
08.org.nvmexpress:uuid:4c4c4544-0056-5410-8048-b9c04f425633
        nvme_3
                  regular   nqn.2014-
08.org.nvmexpress:uuid:4c4c4544-0056-5410-8048-b9c04f425633
4 entries were displayed.
```

If the `hostnqn` strings do not match, use the `vserver modify` command to update the `hostnqn` string on your corresponding ONTAP array subsystem to match the `hostnqn` string from `/etc/nvme/hostnqn` on the host.

Step 3: Configure NVMe/FC and NVMe/TCP

Configure NVMe/FC with Broadcom/Emulex or Marvell/QLogic adapters, or configure NVMe/TCP using manual discovery and connect operations.

NVMe/FC - Broadcom/Emulex

Configure NVMe/FC for a Broadcom/Emulex adapter.

Steps

1. Verify that you are using the supported adapter model:
   a. Display the model names:

```
cat /sys/class/scsi_host/host*/modelname
```

   You should see the following output:

```
LPe36002-M64
LPe36002-M64
```

   b. Display the model descriptions:

```
cat /sys/class/scsi_host/host*/modeldesc
```

   You should see an output similar to the following example:

```
Emulex LightPulse LPe36002-M64 2-Port 64Gb Fibre Channel Adapter
Emulex LightPulse LPe36002-M64 2-Port 64Gb Fibre Channel Adapter
```

2. Verify that you are using the recommended Broadcom `lpfc` firmware and inbox driver:
   a. Display the firmware version:

```
cat /sys/class/scsi_host/host*/fwrev
```

   The command returns the firmware versions:

```
14.0.539.16, sli-4:6:d
14.0.539.16, sli-4:6:d
```

   b. Display the inbox driver version:

```
cat /sys/module/lpfc/version
```

   The following example shows a driver version:

```
0:14.4.0.6
```

For the current list of supported adapter driver and firmware versions, see the Interoperability Matrix Tool.

3. Verify that `lpfc_enable_fc4_type` is set to `3`:

```
cat /sys/module/lpfc/parameters/lpfc_enable_fc4_type
```

4. Verify that you can view your initiator ports:

```
cat /sys/class/fc_host/host*/port_name
```

The following example shows port identities:

```
0x2100f4c7aa0cd7c2
0x2100f4c7aa0cd7c3
```

5. Verify that your initiator ports are online:

```
cat /sys/class/fc_host/host*/port_state
```

You should see the following output:

```
Online
Online
```

6. Verify that the NVMe/FC initiator ports are enabled and that the target ports are visible:

```
cat /sys/class/scsi_host/host*/nvme_info
```

Show example

```
NVME Initiator Enabled
XRI Dist lpfc0 Total 6144 IO 5894 ELS 250
NVME LPORT lpfc0 WWPN x100000109b954518 WWNN x200000109b954518
DID x000000 ONLINE

NVME Statistics
LS: Xmt 0000000000 Cmpl 0000000000 Abort 00000000
LS XMIT: Err 00000000  CMPL: xb 00000000 Err 00000000
Total FCP Cmpl 0000000000000000 Issue 0000000000000000 OutIO
0000000000000000
          abort 00000000 noxri 00000000 nondlp 00000000 qdepth
00000000 wqerr 00000000 err 00000000
FCP CMPL: xb 00000000 Err 00000000

NVME Initiator Enabled
XRI Dist lpfc1 Total 6144 IO 5894 ELS 250
NVME LPORT lpfc1 WWPN x100000109b954519 WWNN x200000109b954519
DID x020500 ONLINE

NVME Statistics
LS: Xmt 0000000000 Cmpl 0000000000 Abort 00000000
LS XMIT: Err 00000000  CMPL: xb 00000000 Err 00000000
Total FCP Cmpl 0000000000000000 Issue 0000000000000000 OutIO
0000000000000000
         abort 00000000 noxri 00000000 nondlp 00000000 qdepth
00000000 wqerr 00000000 err 00000000
FCP CMPL: xb 00000000 Err 00000000

NVME Initiator Enabled
XRI Dist lpfc2 Total 6144 IO 5894 ELS 250
NVME LPORT lpfc2 WWPN x100000109bf044b1 WWNN x200000109bf044b1
DID x022a00 ONLINE
NVME RPORT       WWPN x200bd039eaa7dfc8 WWNN x2008d039eaa7dfc8
DID x021319 TARGET DISCSRVC ONLINE
NVME RPORT       WWPN x2155d039eaa7dfc8 WWNN x2154d039eaa7dfc8
DID x02130f TARGET DISCSRVC ONLINE
NVME RPORT       WWPN x2001d039eaa7dfc8 WWNN x2000d039eaa7dfc8
DID x021310 TARGET DISCSRVC ONLINE
NVME RPORT       WWPN x200dd039eaa7dfc8 WWNN x2008d039eaa7dfc8
DID x020b15 TARGET DISCSRVC ONLINE
NVME RPORT       WWPN x2156d039eaa7dfc8 WWNN x2154d039eaa7dfc8
DID x020b0d TARGET DISCSRVC ONLINE
NVME RPORT       WWPN x2003d039eaa7dfc8 WWNN x2000d039eaa7dfc8
DID x020b10 TARGET DISCSRVC ONLINE
```

```
NVME Statistics
LS: Xmt 0000003049 Cmpl 0000003049 Abort 00000000
LS XMIT: Err 00000000  CMPL: xb 00000000 Err 00000000
Total FCP Cmpl 0000000018f9450b Issue 0000000018f5de57 OutIO
fffffffffffc994c
          abort 000036d3 noxri 00000313 nondlp 00000c8d qdepth
00000000 wqerr 00000064 err 00000000
FCP CMPL: xb 000036d1 Err 000fef0f

NVME Initiator Enabled
XRI Dist lpfc3 Total 6144 IO 5894 ELS 250
NVME LPORT lpfc3 WWPN x100000109bf044b2 WWNN x200000109bf044b2
DID x021b00 ONLINE
NVME RPORT       WWPN x2062d039eaa7dfc8 WWNN x2008d039eaa7dfc8
DID x022915 TARGET DISCSRVC ONLINE
NVME RPORT       WWPN x2157d039eaa7dfc8 WWNN x2154d039eaa7dfc8
DID x02290f TARGET DISCSRVC ONLINE
NVME RPORT       WWPN x2002d039eaa7dfc8 WWNN x2000d039eaa7dfc8
DID x022910 TARGET DISCSRVC ONLINE
NVME RPORT       WWPN x2065d039eaa7dfc8 WWNN x2008d039eaa7dfc8
DID x020119 TARGET DISCSRVC ONLINE
NVME RPORT       WWPN x2158d039eaa7dfc8 WWNN x2154d039eaa7dfc8
DID x02010d TARGET DISCSRVC ONLINE
NVME RPORT       WWPN x2004d039eaa7dfc8 WWNN x2000d039eaa7dfc8
DID x020110 TARGET DISCSRVC ONLINE

NVME Statistics
LS: Xmt 0000002f2c Cmpl 0000002f2c Abort 00000000
LS XMIT: Err 00000000  CMPL: xb 00000000 Err 00000000
Total FCP Cmpl 000000001aaf3eb5 Issue 000000001aab4373 OutIO
fffffffffffc04be
          abort 000035cc noxri 0000038c nondlp 000009e3 qdepth
00000000 wqerr 00000082 err 00000000
FCP CMPL: xb 000035cc Err 000fcfc0
```

NVMe/FC - Marvell/QLogic

Configure NVMe/FC for a Marvell/QLogic adapter.

Steps

1. Verify that you are using the supported adapter driver and firmware versions:

```
cat /sys/class/fc_host/host*/symbolic_name
```

The following example shows driver and firmware versions:

```
QLE2872 FW:v9.15.00 DVR:v10.02.09.300-k
QLE2872 FW:v9.15.00 DVR:v10.02.09.300-k
```

2. Verify that `ql2xnvmeenable` is set. This enables the Marvell adapter to function as an NVMe/FC initiator:

```
cat /sys/module/qla2xxx/parameters/ql2xnvmeenable
```

The expected output is 1.

NVMe/TCP

The NVMe/TCP protocol does not support the auto-connect operation. You must manually perform the NVMe/TCP connect or connect-all operations to discover the NVMe/TCP subsystems and namespaces.

Steps

1. Check that the initiator port can get the discovery log page data across the supported NVMe/TCP LIFs:

```
nvme discover -t tcp -w host-traddr -a traddr
```

Show example

```
nvme discover -t tcp -w 192.168.1.31 -a 192.168.1.24

Discovery Log Number of Records 20, Generation counter 25
=====Discovery Log Entry 0======
trtype:  tcp
adrfam:  ipv4
subtype: current discovery subsystem
treq:    not specified
portid:  4
trsvcid: 8009
subnqn:  nqn.1992-
08.com.netapp:sn.0f4ba1e74eb611ef9f50d039eab6cb6d:discovery
traddr:  192.168.2.25
eflags:  explicit discovery connections, duplicate discovery
information
sectype: none
=====Discovery Log Entry 1======
trtype:  tcp
adrfam:  ipv4
subtype: current discovery subsystem
treq:    not specified
portid:  2
trsvcid: 8009
subnqn:  nqn.1992-
08.com.netapp:sn.0f4ba1e74eb611ef9f50d039eab6cb6d:discovery
traddr:  192.168.1.25
eflags:  explicit discovery connections, duplicate discovery
information
sectype: none
=====Discovery Log Entry 2======
trtype:  tcp
adrfam:  ipv4
subtype: current discovery subsystem
treq:    not specified
portid:  5
trsvcid: 8009
subnqn:  nqn.1992-
08.com.netapp:sn.0f4ba1e74eb611ef9f50d039eab6cb6d:discovery
traddr:  192.168.2.24
eflags:  explicit discovery connections, duplicate discovery
information
sectype: none
=====Discovery Log Entry 3======
trtype:  tcp
```

```
adrfam:  ipv4
subtype: current discovery subsystem
treq:    not specified
portid:  1
trsvcid: 8009
subnqn:  nqn.1992-
08.com.netapp:sn.0f4ba1e74eb611ef9f50d039eab6cb6d:discovery
traddr:  192.168.1.24
eflags:  explicit discovery connections, duplicate discovery
information
sectype: none
=====Discovery Log Entry 4======
trtype:  tcp
adrfam:  ipv4
subtype: nvme subsystem
treq:    not specified
portid:  4
trsvcid: 4420
subnqn:  nqn.1992-
08.com.netapp:sn.0f4ba1e74eb611ef9f50d039eab6cb6d:subsystem.
Unidirectional_DHCP_NONE_1_3
traddr:  192.168.2.25
eflags:  none
sectype: none
=====Discovery Log Entry 5======
trtype:  tcp
adrfam:  ipv4
subtype: nvme subsystem
treq:    not specified
portid:  2
trsvcid: 4420
subnqn:  nqn.1992-
08.com.netapp:sn.0f4ba1e74eb611ef9f50d039eab6cb6d:subsystem.
Unidirectional_DHCP_NONE_1_4
traddr:  192.168.1.25
eflags:  none
sectype: none
=====Discovery Log Entry 6======
trtype:  tcp
adrfam:  ipv4
subtype: nvme subsystem
treq:    not specified
portid:  5
trsvcid: 4420
subnqn:  nqn.1992-
08.com.netapp:sn.0f4ba1e74eb611ef9f50d039eab6cb6d:subsystem.
```

```
Unidirectional_DHCP_NONE_1_5
traddr:  192.168.2.24
eflags:  none
sectype: none
=====Discovery Log Entry 7======
trtype:  tcp
adrfam:  ipv4
subtype: nvme subsystem
treq:    not specified
portid:  1
trsvcid: 4420
subnqn:  nqn.1992-
08.com.netapp:sn.0f4ba1e74eb611ef9f50d039eab6cb6d:subsystem.
Unidirectional_DHCP_2_2
traddr:  192.168.1.24
eflags:  none
sectype: none
=====Discovery Log Entry 8======
trtype:  tcp
adrfam:  ipv4
subtype: nvme subsystem
treq:    not specified
portid:  4
trsvcid: 4420
subnqn:  nqn.1992-
08.com.netapp:sn.0f4ba1e74eb611ef9f50d039eab6cb6d:subsystem.
Unidirectional_DHCP_2_3
traddr:  192.168.2.25
eflags:  none
sectype: none
=====Discovery Log Entry 9======
trtype:  tcp
adrfam:  ipv4
subtype: nvme subsystem
treq:    not specified
portid:  2
trsvcid: 4420
subnqn:  nqn.1992-
08.com.netapp:sn.0f4ba1e74eb611ef9f50d039eab6cb6d:subsystem.
Unidirectional_DHCP_2_5
traddr:  192.168.1.25
eflags:  none
sectype: none
=====Discovery Log Entry 10======
trtype:  tcp
adrfam:  ipv4
```

```
subtype: nvme subsystem
treq:    not specified
portid:  5
trsvcid: 4420
subnqn:  nqn.1992-
08.com.netapp:sn.0f4ba1e74eb611ef9f50d039eab6cb6d:subsystem.
Bidirectional_DHCP_2_2
traddr:  192.168.2.24
eflags:  none
sectype: none
=====Discovery Log Entry 11======
trtype:  tcp
adrfam:  ipv4
subtype: nvme subsystem
treq:    not specified
portid:  1
trsvcid: 4420
subnqn:  nqn.1992-
08.com.netapp:sn.0f4ba1e74eb611ef9f50d039eab6cb6d:subsystem.
Bidirectional_DHCP_2_3
traddr:  192.168.1.24
eflags:  none
sectype: none
=====Discovery Log Entry 12======
trtype:  tcp
adrfam:  ipv4
subtype: nvme subsystem
treq:    not specified
portid:  4
trsvcid: 4420
subnqn:  nqn.1992-
08.com.netapp:sn.0f4ba1e74eb611ef9f50d039eab6cb6d:subsystem.
Bidirectional_DHCP_2_3
traddr:  192.168.2.25
eflags:  none
sectype: none
=====Discovery Log Entry 13======
trtype:  tcp
adrfam:  ipv4
subtype: nvme subsystem
treq:    not specified
portid:  2
trsvcid: 4420
subnqn:  nqn.1992-
08.com.netapp:sn.0f4ba1e74eb611ef9f50d039eab6cb6d:subsystem.
Bidirectional_DHCP_NONE_2_4
```

```
traddr:  192.168.1.25
eflags:  none
sectype: none
=====Discovery Log Entry 14======
trtype:  tcp
adrfam:  ipv4
subtype: nvme subsystem
treq:    not specified
portid:  5
trsvcid: 4420
subnqn:  nqn.1992-
08.com.netapp:sn.0f4ba1e74eb611ef9f50d039eab6cb6d:subsystem.
Bidirectional_DHCP_NONE_2_5
traddr:  192.168.2.24
eflags:  none
sectype: none
=====Discovery Log Entry 15======
trtype:  tcp
adrfam:  ipv4
subtype: nvme subsystem
treq:    not specified
portid:  1
trsvcid: 4420
subnqn:  nqn.1992-
08.com.netapp:sn.0f4ba1e74eb611ef9f50d039eab6cb6d:subsystem.
Bidirectional_DHCP_NONE_2_6
traddr:  192.168.1.24
eflags:  none
sectype: none
=====Discovery Log Entry 16======
trtype:  tcp
adrfam:  ipv4
subtype: nvme subsystem
treq:    not specified
portid:  4
trsvcid: 4420
subnqn:  nqn.1992-
08.com.netapp:sn.0f4ba1e74eb611ef9f50d039eab6cb6d:subsystem.
Bidirectional_DHCP_NONE_2_7
traddr:  192.168.2.25
eflags:  none
sectype: none
=====Discovery Log Entry 17======
trtype:  tcp
adrfam:  ipv4
subtype: nvme subsystem
```

```
treq:    not specified
portid:  2
trsvcid: 4420
subnqn:  nqn.1992-
08.com.netapp:sn.0f4ba1e74eb611ef9f50d039eab6cb6d:subsystem.
Bidirectional_DHCP_NONE_2_8
traddr:  192.168.1.25
eflags:  none
sectype: none
=====Discovery Log Entry 18======
trtype:  tcp
adrfam:  ipv4
subtype: nvme subsystem
treq:    not specified
portid:  5
trsvcid: 4420
subnqn:  nqn.1992-
08.com.netapp:sn.0f4ba1e74eb611ef9f50d039eab6cb6d:subsystem.nvme
_tcp_2
traddr:  192.168.2.24
eflags:  none
sectype: none
=====Discovery Log Entry 19======
trtype:  tcp
adrfam:  ipv4
subtype: nvme subsystem
treq:    not specified
portid:  1
trsvcid: 4420
subnqn:  nqn.1992-
08.com.netapp:sn.0f4ba1e74eb611ef9f50d039eab6cb6d:subsystem.
Bidirectional_DHCP_NONE_2_9
traddr:  192.168.1.24
eflags:  none
sectype: none
```

2. Verify that the other NVMe/TCP initiator-target LIF combinations can successfully get discovery log page data:

```
nvme discover -t tcp -w host-traddr -a traddr
```

Show example

```
nvme discover -t tcp -w 192.168.1.31 -a 192.168.1.24
nvme discover -t tcp -w 192.168.2.31 -a 192.168.2.24
nvme discover -t tcp -w 192.168.1.31 -a 192.168.1.25
nvme discover -t tcp -w 192.168.2.31 -a 192.168.2.25
```

3. Run the `nvme connect-all` command across all the supported NVMe/TCP initiator-target LIFs across the nodes:

```
nvme connect-all -t tcp -w host-traddr -a traddr
```

Show example

```
nvme    connect-all -t  tcp -w  192.168.1.31    -a  192.168.1.24
nvme    connect-all -t  tcp -w  192.168.2.31    -a  192.168.2.24
nvme    connect-all -t  tcp -w  192.168.1.31    -a  192.168.1.25
nvme    connect-all -t  tcp -w  192.168.2.31    -a  192.168.2.25
```

Beginning with Rocky Linux 9.4, the setting for the NVMe/TCP `ctrl_loss_tmo timeout` is automatically set to "off". As a result:

- There are no limits on the number of retries (indefinite retry).
- You don't need to manually configure a specific `ctrl_loss_tmo timeout` duration when using the `nvme connect` or `nvme connect-all` commands (option -l).
- The NVMe/TCP controllers don't experience timeouts in the event of a path failure and remain connected indefinitely.

Step 4: Optionally, enable 1MB I/O for NVMe/FC

ONTAP reports a Max Data Transfer Size (MDTS) of 8 in the Identify Controller data. This means the maximum I/O request size can be up to 1MB. To issue I/O requests of size 1MB for a Broadcom NVMe/FC host, you should increase the `lpfc` value of the `lpfc_sg_seg_cnt` parameter to 256 from the default value of 64.

These steps don't apply to Qlogic NVMe/FC hosts.

Steps

1. Set the `lpfc_sg_seg_cnt` parameter to 256:

```
cat /etc/modprobe.d/lpfc.conf
```

You should see an output similar to the following example:

```
options lpfc lpfc_sg_seg_cnt=256
```

2. Run the `dracut -f` command, and reboot the host.
3. Verify that the value for `lpfc_sg_seg_cnt` is 256:

```
cat /sys/module/lpfc/parameters/lpfc_sg_seg_cnt
```

Step 5: Verify NVMe boot services

The `nvmefc-boot-connections.service` and `nvmf-autoconnect.service` boot services included in the NVMe/FC `nvme-cli` package are automatically enabled when the system boots.

After booting completes, verify that the `nvmefc-boot-connections.service` and `nvmf-autoconnect.service` boot services are enabled.

Steps

1. Verify that `nvmf-autoconnect.service` is enabled:

```
systemctl status nvmf-autoconnect.service
```

Show example output

```
nvmf-autoconnect.service - Connect NVMe-oF subsystems automatically
during boot
     Loaded: loaded (/usr/lib/systemd/system/nvmf-
autoconnect.service; enabled; preset: disabled)
     Active: inactive (dead)

Jun 10 04:06:26 SR630-13-201.lab.eng.btc.netapp.in systemd[1]:
Starting Connect NVMe-oF subsystems automatically during boot...
Jun 10 04:06:26 SR630-13-201.lab.eng.btc.netapp.in systemd[1]: nvmf-
autoconnect.service: Deactivated successfully.
Jun 10 04:06:26 SR630-13-201.lab.eng.btc.netapp.in systemd[1]:
Finished Connect NVMe-oF subsystems automatically during boot.
```

2. Verify that `nvmefc-boot-connections.service` is enabled:

```
systemctl status nvmefc-boot-connections.service
```

Show example output

```
nvmefc-boot-connections.service - Auto-connect to subsystems on FC-
NVME devices found during boot
     Loaded: loaded (/usr/lib/systemd/system/nvmefc-boot-
connections.service; enabled; preset: enabled)
     Active: inactive (dead) since Tue 2025-06-10 01:08:36 EDT; 2h
59min ago
   Main PID: 7090 (code=exited, status=0/SUCCESS)
        CPU: 30ms

Jun 10 01:08:36 localhost systemd[1]: Starting Auto-connect to
subsystems on FC-NVME devices found during boot...
Jun 10 01:08:36 localhost systemd[1]: nvmefc-boot-
connections.service: Deactivated successfully.
Jun 10 01:08:36 localhost systemd[1]: Finished Auto-connect to
subsystems on FC-NVME devices found during boot.
```

Step 6: Verify the multipathing configuration

Verify that the in-kernel NVMe multipath status, ANA status, and ONTAP namespaces are correct for the NVMe-oF configuration.

Steps

1. Verify that the in-kernel NVMe multipath is enabled:

   ```
   cat /sys/module/nvme_core/parameters/multipath
   ```

   You should see the following output:

   ```
   Y
   ```

2. Verify that the appropriate NVMe-oF settings (such as, model set to NetApp ONTAP Controller and load balancing iopolicy set to round-robin) for the respective ONTAP namespaces correctly reflect on the host:

   a. Display the subsystems:

      ```
      cat /sys/class/nvme-subsystem/nvme-subsys*/model
      ```

      You should see the following output:

```
NetApp ONTAP Controller
NetApp ONTAP Controller
```

b. Display the policy:

```
cat /sys/class/nvme-subsystem/nvme-subsys*/iopolicy
```

You should see the following output:

```
round-robin
round-robin
```

3. Verify that the namespaces are created and correctly discovered on the host:

```
nvme list
```

Show example

```
Node         SN                   Model
------------------------------------------------------------
/dev/nvme4n1 81Ix2BVuekWcAAAAAAAB   NetApp ONTAP Controller


Namespace Usage    Format             FW             Rev
------------------------------------------------------------
1                 21.47 GB / 21.47 GB   4 KiB + 0 B   FFFFFFFF
```

4. Verify that the controller state of each path is live and has the correct ANA status:

NVMe/FC

```
nvme list-subsys /dev/nvme4n5
```

Show example

```
nvme-subsys4 - NQN=nqn.1992-
08.com.netapp:sn.3a5d31f5502c11ef9f50d039eab6cb6d:subsystem.nvme
_1
               hostnqn=nqn.2014-08.org.nvmexpress:uuid:e6dade64-
216d-
11ec-b7bb-7ed30a5482c3
iopolicy=round-robin\
+- nvme1 fc traddr=nn-0x2082d039eaa7dfc8:pn-
0x2088d039eaa7dfc8,host_traddr=nn-0x20000024ff752e6d:pn-
0x21000024ff752e6d live optimized
+- nvme12 fc traddr=nn-0x2082d039eaa7dfc8:pn-
0x208ad039eaa7dfc8,host_traddr=nn-0x20000024ff752e6d:pn-
0x21000024ff752e6d live non-optimized
+- nvme10 fc traddr=nn-0x2082d039eaa7dfc8:pn-
0x2087d039eaa7dfc8,host_traddr=nn-0x20000024ff752e6c:pn-
0x21000024ff752e6c live non-optimized
+- nvme3 fc traddr=nn-0x2082d039eaa7dfc8:pn-
0x2083d039eaa7dfc8,host_traddr=nn-0x20000024ff752e6c:pn-
0x21000024ff752e6c live optimized
```

NVMe/TCP

```
nvme list-subsys /dev/nvme1n1
```

Show example

```
nvme-subsys5 - NQN=nqn.1992-
08.com.netapp:sn.0f4ba1e74eb611ef9f50d039eab6cb6d:subsystem.nvme
_tcp_3
hostnqn=nqn.2014-08.org.nvmexpress:uuid:4c4c4544-0035-5910-804b-
b5c04f444d33
iopolicy=round-robin
\
+- nvme13 tcp
traddr=192.168.2.25,trsvcid=4420,host_traddr=192.168.2.31,
src_addr=192.168.2.31 live optimized
+- nvme14 tcp
traddr=192.168.2.24,trsvcid=4420,host_traddr=192.168.2.31,
src_addr=192.168.2.31 live non-optimized
+- nvme5 tcp
traddr=192.168.1.25,trsvcid=4420,host_traddr=192.168.1.31,
src_addr=192.168.1.31 live optimized
+- nvme6 tcp
traddr=192.168.1.24,trsvcid=4420,host_traddr=192.168.1.31,
src_addr=192.168.1.31 live non-optimized
```

5. Verify that the NetApp plug-in displays the correct values for each ONTAP namespace device:

Column

```
nvme netapp ontapdevices -o column
```

Show example

```
Device        Vserver   Namespace Path
------------------------ ------------------------------
/dev/nvme1n1     linux_tcnvme_iscsi
/vol/tcpnvme_1_0_0/tcpnvme_ns

NSID       UUID                                   Size
--------------------------------------------------------------
1    5f7f630d-8ea5-407f-a490-484b95b15dd6   21.47GB
```

JSON

```
nvme netapp ontapdevices -o json
```

Show example

```
{
  "ONTAPdevices":[
    {
      "Device":"/dev/nvme1n1",
      "Vserver":"linux_tcnvme_iscsi",
      "Namespace_Path":"/vol/tcpnvme_1_0_0/tcpnvme_ns",
      "NSID":1,
      "UUID":"5f7f630d-8ea5-407f-a490-484b95b15dd6",
      "Size":"21.47GB",
      "LBA_Data_Size":4096,
      "Namespace_Size":5242880
    },
]
}
```

Step 7: Set up secure in-band authentication

Secure in-band authentication is supported over NVMe/TCP between a Rocky Linux 9x host and an ONTAP controller.

Each host or controller must be associated with a `DH-HMAC-CHAP` key to set up secure authentication. A `DH-HMAC-CHAP` key is a combination of the NQN of the NVMe host or controller and an authentication secret configured by the administrator. To authenticate its peer, an NVMe host or controller must recognize the key associated with the peer.

Steps

Set up secure in-band authentication using the CLI or a config JSON file. If you need to specify different dhchap keys for different subsystems, you must use a config JSON file.

CLI

Set up secure in-band authentication using the CLI.

1. Obtain the host NQN:

```
cat /etc/nvme/hostnqn
```

2. Generate the dhchap key for the Rocky Linux 9.x host.

   The following output describes the `gen-dhchap-key` command parameters:

```
nvme gen-dhchap-key -s optional_secret -l key_length {32|48|64} -m
HMAC_function {0|1|2|3} -n host_nqn
•   -s secret key in hexadecimal characters to be used to initialize
the host key
•   -l length of the resulting key in bytes
•   -m HMAC function to use for key transformation
0 = none, 1- SHA-256, 2 = SHA-384, 3=SHA-512
•   -n host NQN to use for key transformation
```

   In the following example, a random dhchap key with HMAC set to 3 (SHA-512) is generated.

```
nvme gen-dhchap-key -m 3 -n nqn.2014-
08.org.nvmexpress:uuid:e6dade64-216d-11ec-b7bb-7ed30a5482c3
DHHC-
1:03:wSpuuKbBHTzC0W9JZxMBsYd9JFV8Si9aDh22k2BR/4m852vH7KGlrJeMpzhmyjD
WOo0PJJM6yZsTeEpGkDHMHQ255+g=:
```

3. On the ONTAP controller, add the host and specify both dhchap keys:

```
vserver nvme subsystem host add -vserver <svm_name> -subsystem
<subsystem> -host-nqn <host_nqn> -dhchap-host-secret
<authentication_host_secret> -dhchap-controller-secret
<authentication_controller_secret> -dhchap-hash-function {sha-
256|sha-512} -dhchap-group {none|2048-bit|3072-bit|4096-bit|6144-
bit|8192-bit}
```

4. A host supports two types of authentication methods, unidirectional and bidirectional. On the host, connect to the ONTAP controller and specify dhchap keys based on the chosen authentication method:

```
nvme connect -t tcp -w <host-traddr> -a <tr-addr> -n <host_nqn> -S
<authentication_host_secret> -C <authentication_controller_secret>
```

5. Validate the `nvme connect authentication` command by verifying the host and controller dhchap keys:
   a. Verify the host dhchap keys:

```
cat /sys/class/nvme-subsystem/<nvme-subsysX>/nvme*/dhchap_secret
```

Show example output for a unidirectional configuration

```
cat /sys/class/nvme-subsystem/nvme-subsys1/nvme*/dhchap_secret
DHHC-1:01:CNxTYq73T9vJk0JpOfDBZrhDCqpWBN4XVZI5WxwPgDUieHAi:
DHHC-1:01:CNxTYq73T9vJk0JpOfDBZrhDCqpWBN4XVZI5WxwPgDUieHAi:
DHHC-1:01:CNxTYq73T9vJk0JpOfDBZrhDCqpWBN4XVZI5WxwPgDUieHAi:
DHHC-1:01:CNxTYq73T9vJk0JpOfDBZrhDCqpWBN4XVZI5WxwPgDUieHAi:
```

   b. Verify the controller dhchap keys:

```
cat /sys/class/nvme-subsystem/<nvme-
subsysX>/nvme*/dhchap_ctrl_secret
```

Show example output for a bidirectional configuration

```
cat /sys/class/nvme-subsystem/nvme-
subsys6/nvme*/dhchap_ctrl_secret
DHHC-
1:03:wSpuuKbBHTzC0W9JZxMBsYd9JFV8Si9aDh22k2BR/4m852vH7KGlrJeMp
zhmyjDWOo0PJJM6yZsTeEpGkDHMHQ255+g=:
DHHC-
1:03:wSpuuKbBHTzC0W9JZxMBsYd9JFV8Si9aDh22k2BR/4m852vH7KGlrJeMp
zhmyjDWOo0PJJM6yZsTeEpGkDHMHQ255+g=:
DHHC-
1:03:wSpuuKbBHTzC0W9JZxMBsYd9JFV8Si9aDh22k2BR/4m852vH7KGlrJeMp
zhmyjDWOo0PJJM6yZsTeEpGkDHMHQ255+g=:
DHHC-
1:03:wSpuuKbBHTzC0W9JZxMBsYd9JFV8Si9aDh22k2BR/4m852vH7KGlrJeMp
zhmyjDWOo0PJJM6yZsTeEpGkDHMHQ255+g=:
```

JSON

When multiple NVMe subsystems are available on the ONTAP controller, you can use the `/etc/nvme/config.json` file with the `nvme connect-all` command.

Use the `-o` option to generate the JSON file. Refer to the NVMe connect-all man pages for more syntax options.

1. Configure the JSON file.

In the following example, `dhchap_key` corresponds to `dhchap_secret` and `dhchap_ctrl_key` corresponds to `dhchap_ctrl_secret`.

Show example

```
cat /etc/nvme/config.json
[
{
  "hostnqn":"nqn.2014-08.org.nvmexpress:uuid:9796c1ec-0d34-11eb-
  b6b2-3a68dd3bab57",
  "hostid":"b033cd4fd6db4724adb48655bfb55448",
  "dhchap_key":" DHHC-
1:01:CNxTYq73T9vJk0JpOfDBZrhDCqpWBN4XVZI5WxwPgDUieHAi:"
},
{
  "hostnqn":"nqn.2014-08.org.nvmexpress:uuid:4c4c4544-0035-5910-
  804b-b5c04f444d33",
  "subsystems":[
        {
          "nqn":"nqn.1992-

08.com.netapp:sn.0f4ba1e74eb611ef9f50d039eab6cb6d:subsystem.bidi
          r_DHCP",
          "ports":[
              {
                  "transport":"tcp",
                    "traddr":" 192.168.1.24 ",
                  "host_traddr":" 192.168.1.31 ",
                  "trsvcid":"4420",
                  "dhchap_ctrl_key":"DHHC-
                  1:03:
wSpuuKbBHTzC0W9JZxMBsYd9JFV8Si9aDh22k2BR/4m852vH7KGlrJeMpzhmyjDW
Oo0PJJM6yZsTeEpGkDHMHQ255+g=:"
              },
              {
                  "transport":"tcp",
                  "traddr":" 192.168.1.25 ",
                  "host_traddr":" 192.168.1.31",
                  "trsvcid":"4420",
                  "dhchap_ctrl_key":"DHHC-
                  1:03:
wSpuuKbBHTzC0W9JZxMBsYd9JFV8Si9aDh22k2BR/4m852vH7KGlrJeMpzhmyjDW
Oo0PJJM6yZsTeEpGkDHMHQ255+g=:"
              },
              {
                  "transport":"tcp",
                 "traddr":" 192.168.2.24 ",
                  "host_traddr":" 192.168.2.31",
                  "trsvcid":"4420",
```

```
                "dhchap_ctrl_key":"DHHC-
                1:03:
wSpuuKbBHTzC0W9JZxMBsYd9JFV8Si9aDh22k2BR/4m852vH7KGlrJeMpzhmyjDW
Oo0PJJM6yZsTeEpGkDHMHQ255+g=:"
            },
            {
                "transport":"tcp",
                "traddr":" 192.168.2.25 ",
                  "host_traddr":" 192.168.2.31",
                "trsvcid":"4420",
                "dhchap_ctrl_key":"DHHC-
                1:03:
wSpuuKbBHTzC0W9JZxMBsYd9JFV8Si9aDh22k2BR/4m852vH7KGlrJeMpzhmyjDW
Oo0PJJM6yZsTeEpGkDHMHQ255+g=:"
            }
        ]
    }
  ]
}
]
```

2. Connect to the ONTAP controller using the config JSON file:

```
nvme connect-all -J /etc/nvme/config.json
```

Show example

```
already connected to hostnqn=nqn.2014-
08.org.nvmexpress:uuid:4c4c4544-0035-5910-804b-
b5c04f444d33,nqn=nqn.1992-
08.com.netapp:sn.8dde3be2cc7c11efb777d039eab6cb6d:subsystem.
bidi
r_DHCP,transport=tcp,traddr=192.168.1.25,trsvcid=4420
already connected to hostnqn=nqn.2014-
08.org.nvmexpress:uuid:4c4c4544-0035-5910-804b-
b5c04f444d33,nqn=nqn.1992-
08.com.netapp:sn.8dde3be2cc7c11efb777d039eab6cb6d:subsystem.
bidi
r_DHCP,transport=tcp,traddr=192.168.2.25,trsvcid=4420
already connected to hostnqn=nqn.2014-
08.org.nvmexpress:uuid:4c4c4544-0035-5910-804b-
b5c04f444d33,nqn=nqn.1992-
08.com.netapp:sn.8dde3be2cc7c11efb777d039eab6cb6d:subsystem.
bidi
r_DHCP,transport=tcp,traddr=192.168.1.24,trsvcid=4420
already connected to hostnqn=nqn.2014-
08.org.nvmexpress:uuid:4c4c4544-0035-5910-804b-
b5c04f444d33,nqn=nqn.1992-
08.com.netapp:sn.8dde3be2cc7c11efb777d039eab6cb6d:subsystem.
bidi
r_DHCP,transport=tcp,traddr=192.168.2.24,trsvcid=4420
```

3. Verify that the dhchap secrets have been enabled for the respective controllers for each subsystem:
   a. Verify the host dhchap keys:

      ```
      cat /sys/class/nvme-subsystem/nvme-subsys0/nvme0/dhchap_secret
      ```

      The following example shows a dhchap key:

      ```
      DHHC-1:01:CNxTYq73T9vJk0JpOfDBZrhDCqpWBN4XVZI5WxwPgDUieHAi:
      ```

   b. Verify the controller dhchap keys:

      ```
      cat /sys/class/nvme-subsystem/nvme-
      subsys0/nvme0/dhchap_ctrl_secret
      ```

      You should see an output similar to the following example:

```
DHHC-
1:03:wSpuuKbBHTzC0W9JZxMBsYd9JFV8Si9aDh22k2BR/4m852vH7KGlrJeMpzhm
yjDWOo0PJJM6yZsTeEpGkDHMHQ255+g=:
```

Step 8: Review the known issues

There are no known issues.

Configure Rocky Linux 8.x for NVMe-oF with ONTAP storage

Rocky Linux hosts support the NVMe over Fibre Channel (NVMe/FC) and NVMe over TCP (NVMe/TCP) protocols with Asymmetric Namespace Access (ANA). ANA provides multipathing functionality equivalent to asymmetric logical unit access (ALUA) in iSCSI and FCP environments.

Learn how to configure NVMe over Fabrics (NVMe-oF) hosts for Rocky Linux 8.x. For more support and feature information, see Rocky Linux ONTAP support and features.

NVMe-oF with Rocky Linux 8.x has the following known limitations:

- SAN booting using the NVMe-oF protocol is not currently supported.
- In-kernel NVMe multipath is disabled by default on NVMe-oF hosts in Rocky Linux 8.x; you must enable it manually.
- NVMe/TCP is available as a technology preview due to known issues.

Step 1: Install Rocky Linux and NVMe software and verify your configuration

To configure your host for NVMe-oF you need to install the host and NVMe software packages, enable multipathing, and verify your host NQN configuration.

Steps

1. Install Rocky Linux 8.x on the server. After the installation is complete, verify that you are running the required Rocky Linux 8.x kernel:

   ```
   uname -r
   ```

   Example Rocky Linux kernel version:

   ```
   5.14.0-570.12.1.el9_6.x86_64
   ```

2. Install the `nvme-cli` package:

   ```
   rpm -qa|grep nvme-cli
   ```

The following example shows an nvme-cli package version:

```
nvme-cli-2.11-5.el9.x86_64
```

3. Install the `libnvme` package:

```
rpm -qa|grep libnvme
```

The following example shows an `libnvme` package version:

```
libnvme-1.11.1-1.el9.x86_64
```

4. On the Rocky Linux host, check the hostnqn string at `/etc/nvme/hostnqn`:

```
cat /etc/nvme/hostnqn
```

The following example shows an `hostnqn` version:

```
nqn.2014-08.org.nvmexpress:uuid:4c4c4544-0056-5410-8048-b9c04f425633
```

5. On the ONTAP system, verify that the `hostnqn` string matches the `hostnqn` string for the corresponding subsystem on the ONTAP array:

```
::> vserver nvme subsystem host show -vserver vs_coexistence_LPE36002
```

Show example

```
Vserver Subsystem Priority  Host NQN
------- --------- --------
-------------------------------------------------
vs_coexistence_LPE36002
        nvme
                  regular   nqn.2014-
08.org.nvmexpress:uuid:4c4c4544-0056-5410-8048-b9c04f425633
        nvme_1
                  regular   nqn.2014-
08.org.nvmexpress:uuid:4c4c4544-0056-5410-8048-b9c04f425633
        nvme_2
                  regular   nqn.2014-
08.org.nvmexpress:uuid:4c4c4544-0056-5410-8048-b9c04f425633
        nvme_3
                  regular   nqn.2014-
08.org.nvmexpress:uuid:4c4c4544-0056-5410-8048-b9c04f425633
4 entries were displayed.
```

If the `hostnqn` strings do not match, use the `vserver modify` command to update the `hostnqn` string on your corresponding ONTAP array subsystem to match the `hostnqn` string from `/etc/nvme/hostnqn` on the host.

Step 2: Configure NVMe/FC and NVMe/TCP

Configure NVMe/FC with Broadcom/Emulex or Marvell/QLogic adapters, or configure NVMe/TCP using manual discovery and connect operations.

NVMe/FC - Broadcom/Emulex

Configure NVMe/FC for a Broadcom/Emulex adapter.

Steps

1. Verify that you are using the supported adapter model:
   a. Display the model names:

```
cat /sys/class/scsi_host/host*/modelname
```

   You should see the following output:

```
LPe36002-M64
LPe36002-M64
```

   b. Display the model descriptions:

```
cat /sys/class/scsi_host/host*/modeldesc
```

   You should see an output similar to the following example:

```
Emulex LightPulse LPe36002-M64 2-Port 64Gb Fibre Channel Adapter
Emulex LightPulse LPe36002-M64 2-Port 64Gb Fibre Channel Adapter
```

2. Verify that you are using the recommended Broadcom `lpfc` firmware and inbox driver:
   a. Display the firmware version:

```
cat /sys/class/scsi_host/host*/fwrev
```

   The command returns the firmware versions:

```
14.4.317.10, sli-4:6:d
14.4.317.10, sli-4:6:d
```

   b. Display the inbox driver version:

```
cat /sys/module/lpfc/version`
```

   The following example shows a driver version:

```
0:14.4.0.2
```

For the current list of supported adapter driver and firmware versions, see the Interoperability Matrix Tool.

3. Verify that the expected output of `lpfc_enable_fc4_type` is set to `3`:

```
cat /sys/module/lpfc/parameters/lpfc_enable_fc4_type
```

4. Verify that you can view your initiator ports:

```
cat /sys/class/fc_host/host*/port_name
```

The following example shows port identities:

```
0x100000109bf044b1
0x100000109bf044b2
```

5. Verify that your initiator ports are online:

```
cat /sys/class/fc_host/host*/port_state
```

You should see the following output:

```
Online
Online
```

6. Verify that the NVMe/FC initiator ports are enabled and that the target ports are visible:

```
cat /sys/class/scsi_host/host*/nvme_info
```

Show example

```
NVME Initiator Enabled
XRI Dist lpfc2 Total 6144 IO 5894 ELS 250
NVME LPORT lpfc2 WWPN x100000109bf044b1 WWNN x200000109bf044b1
DID x022a00 ONLINE
NVME RPORT       WWPN x202fd039eaa7dfc8 WWNN x202cd039eaa7dfc8
DID x021310 TARGET DISCSRVC ONLINE
NVME RPORT       WWPN x202dd039eaa7dfc8 WWNN x202cd039eaa7dfc8
DID x020b10 TARGET DISCSRVC ONLINE

NVME Statistics
LS: Xmt 0000000810 Cmpl 0000000810 Abort 00000000
LS XMIT: Err 00000000  CMPL: xb 00000000 Err 00000000
Total FCP Cmpl 000000007b098f07 Issue 000000007aee27c4 OutIO
fffffffffe498bd
        abort 000013b4 noxri 00000000 nondlp 00000058 qdepth
00000000 wqerr 00000000 err 00000000
FCP CMPL: xb 000013b4 Err 00021443

NVME Initiator Enabled
XRI Dist lpfc3 Total 6144 IO 5894 ELS 250
NVME LPORT lpfc3 WWPN x100000109bf044b2 WWNN x200000109bf044b2
DID x021b00 ONLINE
NVME RPORT       WWPN x2033d039eaa7dfc8 WWNN x202cd039eaa7dfc8
DID x020110 TARGET DISCSRVC ONLINE
NVME RPORT       WWPN x2032d039eaa7dfc8 WWNN x202cd039eaa7dfc8
DID x022910 TARGET DISCSRVC ONLINE

NVME Statistics
LS: Xmt 0000000840 Cmpl 0000000840 Abort 00000000
LS XMIT: Err 00000000  CMPL: xb 00000000 Err 00000000
Total FCP Cmpl 000000007afd4434 Issue 000000007ae31b83 OutIO
fffffffffe5d74f
        abort 000014a5 noxri 00000000 nondlp 0000006a qdepth
00000000 wqerr 00000000 err 00000000
FCP CMPL: xb 000014a5 Err 0002149a
```

NVMe/FC - Marvell/QLogic

Configure NVMe/FC for a Marvell/QLogic adapter.

Steps

1. Verify that you are running the supported adapter driver and firmware versions:

```
cat /sys/class/fc_host/host*/symbolic_name
```

The follow example shows driver and firmware versions:

```
QLE2742 FW:v9.14.00 DVR:v10.02.09.200-k
QLE2742 FW:v9.14.00 DVR:v10.02.09.200-k
```

2. Verify that `ql2xnvmeenable` is set. This enables the Marvell adapter to function as an NVMe/FC initiator:

```
cat /sys/module/qla2xxx/parameters/ql2xnvmeenable
```

The expected output is 1.

NVMe/TCP

The NVMe/TCP protocol doesn't support the auto-connect operation. Instead, you can discover the NVMe/TCP subsystems and namespaces by performing the NVMe/TCP `connect` or `connect-all` operations manually.

Steps

1. Check that the initiator port can get the discovery log page data across the supported NVMe/TCP LIFs:

```
nvme discover -t tcp -w host-traddr -a traddr
```

Show example

```
nvme discover -t tcp -w 192.168.1.31 -a 192.168.1.24
Discovery Log Number of Records 20, Generation counter 25
=====Discovery Log Entry 0======
trtype:  tcp
adrfam:  ipv4
subtype: current discovery subsystem
treq:    not specified
portid:  4
trsvcid: 8009
subnqn:  nqn.1992-
08.com.netapp:sn.0f4ba1e74eb611ef9f50d039eab6cb6d:discovery
traddr:  192.168.2.25
eflags:  explicit discovery connections, duplicate discovery
information
sectype: none
=====Discovery Log Entry 1======
trtype:  tcp
adrfam:  ipv4
subtype: current discovery subsystem
treq:    not specified
portid:  2
trsvcid: 8009
subnqn:  nqn.1992-
08.com.netapp:sn.0f4ba1e74eb611ef9f50d039eab6cb6d:discovery
traddr:  192.168.1.25
eflags:  explicit discovery connections, duplicate discovery
information
sectype: none
=====Discovery Log Entry 2======
trtype:  tcp
adrfam:  ipv4
subtype: current discovery subsystem
treq:    not specified
portid:  5
trsvcid: 8009
subnqn:  nqn.1992-
08.com.netapp:sn.0f4ba1e74eb611ef9f50d039eab6cb6d:discovery
traddr:  192.168.2.24
eflags:  explicit discovery connections, duplicate discovery
information
sectype: none
=====Discovery Log Entry 3======
trtype:  tcp
adrfam:  ipv4
```

```
subtype: current discovery subsystem
treq:    not specified
portid:  1
trsvcid: 8009
subnqn:  nqn.1992-
08.com.netapp:sn.0f4ba1e74eb611ef9f50d039eab6cb6d:discovery
traddr:  192.168.1.24
eflags:  explicit discovery connections, duplicate discovery
information
sectype: none
=====Discovery Log Entry 4======
trtype:  tcp
adrfam:  ipv4
subtype: nvme subsystem
treq:    not specified
portid:  4
trsvcid: 4420
subnqn:  nqn.1992-
08.com.netapp:sn.0f4ba1e74eb611ef9f50d039eab6cb6d:subsystem.nvme
_tcp_1
traddr:  192.168.2.25
eflags:  none
sectype: none
=====Discovery Log Entry 5======
trtype:  tcp
adrfam:  ipv4
subtype: nvme subsystem
treq:    not specified
portid:  2
trsvcid: 4420
subnqn:  nqn.1992-
08.com.netapp:sn.0f4ba1e74eb611ef9f50d039eab6cb6d:subsystem.nvme
_tcp_1
traddr:  192.168.1.25
eflags:  none
sectype: none
=====Discovery Log Entry 6======
trtype:  tcp
adrfam:  ipv4
subtype: nvme subsystem
treq:    not specified
portid:  5
trsvcid: 4420
subnqn:  nqn.1992-
08.com.netapp:sn.0f4ba1e74eb611ef9f50d039eab6cb6d:subsystem.nvme
_tcp_1
```

```
traddr:  192.168.2.24
eflags:  none
sectype: none
=====Discovery Log Entry 7======
trtype:  tcp
adrfam:  ipv4
subtype: nvme subsystem
treq:    not specified
portid:  1
trsvcid: 4420
subnqn:  nqn.1992-
08.com.netapp:sn.0f4ba1e74eb611ef9f50d039eab6cb6d:subsystem.nvme
_tcp_1
traddr:  192.168.1.24
eflags:  none
sectype: none
=====Discovery Log Entry 8======
trtype:  tcp
adrfam:  ipv4
subtype: nvme subsystem
treq:    not specified
portid:  4
trsvcid: 4420
subnqn:  nqn.1992-
08.com.netapp:sn.0f4ba1e74eb611ef9f50d039eab6cb6d:subsystem.nvme
_tcp_4
traddr:  192.168.2.25
eflags:  none
sectype: none
=====Discovery Log Entry 9======
trtype:  tcp
adrfam:  ipv4
subtype: nvme subsystem
treq:    not specified
portid:  2
trsvcid: 4420
subnqn:  nqn.1992-
08.com.netapp:sn.0f4ba1e74eb611ef9f50d039eab6cb6d:subsystem.nvme
_tcp_4
traddr:  192.168.1.25
eflags:  none
sectype: none
=====Discovery Log Entry 10======
trtype:  tcp
adrfam:  ipv4
subtype: nvme subsystem
```

```
treq:    not specified
portid:  5
trsvcid: 4420
subnqn:  nqn.1992-
08.com.netapp:sn.0f4ba1e74eb611ef9f50d039eab6cb6d:subsystem.nvme
_tcp_4
traddr:  192.168.2.24
eflags:  none
sectype: none
=====Discovery Log Entry 11======
trtype:  tcp
adrfam:  ipv4
subtype: nvme subsystem
treq:    not specified
portid:  1
trsvcid: 4420
subnqn:  nqn.1992-
08.com.netapp:sn.0f4ba1e74eb611ef9f50d039eab6cb6d:subsystem.nvme
_tcp_4
traddr:  192.168.1.24
eflags:  none
sectype: none
=====Discovery Log Entry 12======
trtype:  tcp
adrfam:  ipv4
subtype: nvme subsystem
treq:    not specified
portid:  4
trsvcid: 4420
subnqn:  nqn.1992-
08.com.netapp:sn.0f4ba1e74eb611ef9f50d039eab6cb6d:subsystem.nvme
_tcp_3
traddr:  192.168.2.25
eflags:  none
sectype: none
=====Discovery Log Entry 13======
trtype:  tcp
adrfam:  ipv4
subtype: nvme subsystem
treq:    not specified
portid:  2
trsvcid: 4420
subnqn:  nqn.1992-
08.com.netapp:sn.0f4ba1e74eb611ef9f50d039eab6cb6d:subsystem.nvme
_tcp_3
traddr:  192.168.1.25
```

```
eflags:  none
sectype: none
=====Discovery Log Entry 14======
trtype:  tcp
adrfam:  ipv4
subtype: nvme subsystem
treq:    not specified
portid:  5
trsvcid: 4420
subnqn:  nqn.1992-
08.com.netapp:sn.0f4ba1e74eb611ef9f50d039eab6cb6d:subsystem.nvme
_tcp_3
traddr:  192.168.2.24
eflags:  none
sectype: none
=====Discovery Log Entry 15======
trtype:  tcp
adrfam:  ipv4
subtype: nvme subsystem
treq:    not specified
portid:  1
trsvcid: 4420
subnqn:  nqn.1992-
08.com.netapp:sn.0f4ba1e74eb611ef9f50d039eab6cb6d:subsystem.nvme
_tcp_3
traddr:  192.168.1.24
eflags:  none
sectype: none
=====Discovery Log Entry 16======
trtype:  tcp
adrfam:  ipv4
subtype: nvme subsystem
treq:    not specified
portid:  4
trsvcid: 4420
subnqn:  nqn.1992-
08.com.netapp:sn.0f4ba1e74eb611ef9f50d039eab6cb6d:subsystem.nvme
_tcp_2
traddr:  192.168.2.25
eflags:  none
sectype: none
=====Discovery Log Entry 17======
trtype:  tcp
adrfam:  ipv4
subtype: nvme subsystem
treq:    not specified
```

```
portid:  2
trsvcid: 4420
subnqn:  nqn.1992-
08.com.netapp:sn.0f4ba1e74eb611ef9f50d039eab6cb6d:subsystem.nvme
_tcp_2
traddr:  192.168.1.25
eflags:  none
sectype: none
=====Discovery Log Entry 18======
trtype:  tcp
adrfam:  ipv4
subtype: nvme subsystem
treq:    not specified
portid:  5
trsvcid: 4420
subnqn:  nqn.1992-
08.com.netapp:sn.0f4ba1e74eb611ef9f50d039eab6cb6d:subsystem.nvme
_tcp_2
traddr:  192.168.2.24
eflags:  none
sectype: none
=====Discovery Log Entry 19======
trtype:  tcp
adrfam:  ipv4
subtype: nvme subsystem
treq:    not specified
portid:  1
trsvcid: 4420
subnqn:  nqn.1992-
08.com.netapp:sn.0f4ba1e74eb611ef9f50d039eab6cb6d:subsystem.nvme
_tcp_2
traddr:  192.168.1.24
eflags:  none
sectype: none
```

2. Verify that the other NVMe/TCP initiator-target LIF combinations are able to successfully fetch discovery log page data:

```
nvme discover -t tcp -w host-traddr -a traddr
```

Show example

```
nvme discover -t tcp -w 192.168.1.31 -a 192.168.1.24
nvme discover -t tcp -w 192.168.2.31 -a 192.168.2.24
nvme discover -t tcp -w 192.168.1.31 -a 192.168.1.25
nvme discover -t tcp -w 192.168.2.31 -a 192.168.2.25
```

3. Run the `nvme connect-all` command across all the supported NVMe/TCP initiator-target LIFs across the nodes:

```
nvme connect-all -t tcp -w host-traddr -a traddr
```

Show example

```
nvme    connect-all -t  tcp -w  192.168.1.31    -a  192.168.1.24
nvme    connect-all -t  tcp -w  192.168.2.31    -a  192.168.2.24
nvme    connect-all -t  tcp -w  192.168.1.31    -a  192.168.1.25
nvme    connect-all -t  tcp -w  192.168.2.31    -a  192.168.2.25
```

Step 3: Optionally, enable 1MB I/O for NVMe/FC

You can enable I/O requests of size 1MB for NVMe/FC configured with a Broadcom adapter. ONTAP reports a Max Data Transfer Size (MDTS) of 8 in the Identify Controller data. This means the maximum I/O request size can be up to 1MB. To issue I/O requests of size 1MB, you need to increase the lpfc value of the `lpfc_sg_seg_cnt` parameter to 256 from the default value of 64.

These steps don't apply to Qlogic NVMe/FC hosts.

Steps

1. Set the `lpfc_sg_seg_cnt` parameter to 256:

```
cat /etc/modprobe.d/lpfc.conf
```

```
options lpfc lpfc_sg_seg_cnt=256
```

2. Run the `dracut -f` command, and reboot the host.
3. Verify that the value for `lpfc_sg_seg_cnt` is 256:

```
cat /sys/module/lpfc/parameters/lpfc_sg_seg_cnt
```

Step 4: Verify the multipathing configuration

Verify that the in-kernel NVMe multipath status, ANA status, and ONTAP namespaces are correct for the NVMe-oF configuration.

Steps

1. Verify that the in-kernel NVMe multipath is enabled:

   ```
   cat /sys/module/nvme_core/parameters/multipath
   ```

   You should see the following output:

   ```
   Y
   ```

2. Verify that the appropriate NVMe-oF settings (such as, model set to NetApp ONTAP Controller and load balancing iopolicy set to round-robin) for the respective ONTAP namespaces correctly reflect on the host:

   a. Display the subsystems:

      ```
      cat /sys/class/nvme-subsystem/nvme-subsys*/model
      ```

      You should see the following output:

      ```
      NetApp ONTAP Controller
      NetApp ONTAP Controller
      ```

   b. Display the policy:

      ```
      cat /sys/class/nvme-subsystem/nvme-subsys*/iopolicy
      ```

      You should see the following output:

      ```
      round-robin
      round-robin
      ```

3. Verify that the namespaces are created and correctly discovered on the host:

   ```
   nvme list
   ```

Show example

```
Node         SN                    Model
-------------------------------------------------------
/dev/nvme4n1 81Ix2BVuekWcAAAAAAAB   NetApp ONTAP Controller


Namespace Usage    Format              FW             Rev
-------------------------------------------------------------
1                 21.47 GB / 21.47 GB   4 KiB + 0 B   FFFFFFFF
```

4. Verify that the controller state of each path is live and has the correct ANA status:

NVMe/FC

```
nvme list-subsys /dev/nvme4n5
```

Show example

```
nvme-subsys4 - NQN=nqn.1992-
08.com.netapp:sn.3a5d31f5502c11ef9f50d039eab6cb6d:subsystem.nvme
_1
               hostnqn=nqn.2014-08.org.nvmexpress:uuid:e6dade64-
216d-
11ec-b7bb-7ed30a5482c3
iopolicy=round-robin\
+- nvme1 fc traddr=nn-0x2082d039eaa7dfc8:pn-
0x2088d039eaa7dfc8,host_traddr=nn-0x20000024ff752e6d:pn-
0x21000024ff752e6d live optimized
+- nvme12 fc traddr=nn-0x2082d039eaa7dfc8:pn-
0x208ad039eaa7dfc8,host_traddr=nn-0x20000024ff752e6d:pn-
0x21000024ff752e6d live non-optimized
+- nvme10 fc traddr=nn-0x2082d039eaa7dfc8:pn-
0x2087d039eaa7dfc8,host_traddr=nn-0x20000024ff752e6c:pn-
0x21000024ff752e6c live non-optimized
+- nvme3 fc traddr=nn-0x2082d039eaa7dfc8:pn-
0x2083d039eaa7dfc8,host_traddr=nn-0x20000024ff752e6c:pn-
0x21000024ff752e6c live optimized
```

NVMe/TCP

```
nvme list-subsys /dev/nvme1n1
```

Show example

```
nvme-subsys5 - NQN=nqn.1992-
08.com.netapp:sn.0f4ba1e74eb611ef9f50d039eab6cb6d:subsystem.nvme
_tcp_3
hostnqn=nqn.2014-08.org.nvmexpress:uuid:4c4c4544-0035-5910-804b-
b5c04f444d33
iopolicy=round-robin
\
+- nvme13 tcp
traddr=192.168.2.25,trsvcid=4420,host_traddr=192.168.2.31,
src_addr=192.168.2.31 live optimized
+- nvme14 tcp
traddr=192.168.2.24,trsvcid=4420,host_traddr=192.168.2.31,
src_addr=192.168.2.31 live non-optimized
+- nvme5 tcp
traddr=192.168.1.25,trsvcid=4420,host_traddr=192.168.1.31,
src_addr=192.168.1.31 live optimized
+- nvme6 tcp
traddr=192.168.1.24,trsvcid=4420,host_traddr=192.168.1.31,
src_addr=192.168.1.31 live non-optimized
```

5. Verify that the NetApp plug-in displays the correct values for each ONTAP namespace device:

Column

```
nvme netapp ontapdevices -o column
```

Show example

```
Device        Vserver   Namespace Path
--------------------- ------------------------------
/dev/nvme1n1     linux_tcnvme_iscsi
/vol/tcpnvme_1_0_0/tcpnvme_ns

NSID       UUID                                   Size
-------------------------------------------------------------
1    5f7f630d-8ea5-407f-a490-484b95b15dd6   21.47GB
```

JSON

```
nvme netapp ontapdevices -o json
```

Show example

```
{
  "ONTAPdevices":[
    {
      "Device":"/dev/nvme1n1",
      "Vserver":"linux_tcnvme_iscsi",
      "Namespace_Path":"/vol/tcpnvme_1_0_0/tcpnvme_ns",
      "NSID":1,
      "UUID":"5f7f630d-8ea5-407f-a490-484b95b15dd6",
      "Size":"21.47GB",
      "LBA_Data_Size":4096,
      "Namespace_Size":5242880
    },
]
}
```

Step 5: Review the known issues

These are the known issues:

NetApp Bug ID	Title	Description
1479047	Rocky Linux 8.x NVMe-oF hosts create duplicate persistent discovery controllers	On NVMe-oF hosts, you can use the "nvme discover -p" command to create Persistent Discovery Controllers (PDCs). When this command is used, only one PDC should be created per initiator-target combination. However, if you are running Rocky Linux 8.x on an NVMe-oF host, a duplicate PDC is created each time "nvme discover -p" is executed. This leads to unnecessary usage of resources on both the host and the target.

SUSE Linux Enterprise Server

Learn about ONTAP support and features for SUSE Linux Enterprise Server

Features supported for host configuration with NVMe over Fabrics (NVMe-oF) varies based on your version of ONTAP and SUSE Linux Enterprise Server.

Feature	SUSE Linux Enterprise Server host version	ONTAP version
NVMe/TCP reports all optimized paths for ASA r2 systems	16	9.16.1 or later
Transport Layer Security (TLS) 1.3 encryption is supported for NVMe/TCP	15 SP6 or later	9.16.1 or later
Secure in-band authentication is supported over NVMe/TCP between a RHEL host and an ONTAP controller	15 SP4 or later	9.12.1 or later
Persistent discovery controllers (PDCs) are supported using a unique discovery NQN	15 SP4 or later	9.11.1 or later
NVMe/TCP provides namespaces using the native `nvme-cli` package	15 SP4 or later	9.10.1 or later
NVMe and SCSI traffic is supported on the same host using NVMe multipath for NVMe-oF namespaces and dm-multipath for SCSI LUNs	15 SP1 or later	9.4 or later

ONTAP supports the following SAN host features regardless of the ONTAP version running on your system setup.

Feature	SUSE Linux Enterprise Server host version
SAN booting is enabled using the NVMe/FC protocol	15 SP7 or later
Native NVMe multipathing is enabled by default	15 SP1 or later

Feature	SUSE Linux Enterprise Server host version
The `nvme-cli` package includes auto-connect scripts removing the need for third-party scripts	15 SP1 or later

For details on supported configurations, see the Interoperability Matrix Tool.

What's next

If your SUSE Linux Enterprise Server version is …	Learn about …
16	Configuring NVMe for SUSE Linux Enterprise Server 16
15 SPx series	Configuring NVMe for SUSE Linux Enterprise Server 15 SPx

Related information

- Learn about ASA r2 systems
- Learn about managing NVMe protocols

Configure SUSE Linux Enterprise Server 16 for NVMe-oF with ONTAP storage

The SUSE Linux Enterprise Server 16 host supports the NVMe over Fibre Channel (NVMe/FC) and NVMe over TCP (NVMe/TCP) protocols with Asymmetric Namespace Access (ANA). ANA provides multipathing functionality equivalent to asymmetric logical unit access (ALUA) in iSCSI and FCP environments.

Learn how to configure NVMe over Fabrics (NVMe-oF) hosts for SUSE Linux Enterprise Server 16. For more support and feature information, see ONTAP support and features.

NVMe-oF with SUSE Linux Enterprise Server 16 has the following known limitations:

- The `nvme disconnect-all` command disconnects both root and data filesystems and might lead to system instability. Do not issue this on systems booting from SAN over NVMe-TCP or NVMe-FC namespaces.
- NetApp sanlun host utility support isn't available for NVMe-oF. Instead, you can rely on the NetApp plug-in included in the native `nvme-cli` package for all NVMe-oF transports.

Step 1: Optionally, enable SAN booting

You can configure your host to use SAN booting to simplify deployment and improve scalability. Use the Interoperability Matrix Tool to verify that your Linux OS, host bus adapter (HBA), HBA firmware, HBA boot BIOS, and ONTAP version support SAN booting.

Steps

1. Create a NVMe namespace and map it to the host.
2. Enable SAN booting in the server BIOS for the ports to which the SAN boot namespace is mapped.

   For information on how to enable the HBA BIOS, see your vendor-specific documentation.

3. Reboot the host and verify that the OS is up and running.

Step 2: Install SUSE Linux Enterprise Server and NVMe software and verify your configuration

To configure your host for NVMe-oF you need to install the host and NVMe software packages, enable multipathing, and verify your host NQN configuration.

Steps

1. Install SUSE Linux Enterprise Server 16 on the server. After the installation is complete, verify that you are running the specified SUSE Linux Enterprise Server 16 kernel:

   ```
   uname -r
   ```

   Example SUSE Linux Enterprise Server kernel version:

   ```
   6.12.0-160000.6-default
   ```

2. Install the `nvme-cli` package:

   ```
   rpm -qa|grep nvme-cli
   ```

   The following example shows an `nvme-cli` package version:

   ```
   nvme-cli-2.11+29.g35e62868-160000.1.1.x86_64
   ```

3. Install the `libnvme` package:

   ```
   rpm -qa|grep libnvme
   ```

   The following example shows an `libnvme` package version:

   ```
   libnvme1-1.11+17.g6d55624d-160000.1.1.x86_64
   ```

4. On the host, check the hostnqn string at `/etc/nvme/hostnqn`:

   ```
   cat /etc/nvme/hostnqn
   ```

   The following example shows a `hostnqn` version:

```
nqn.2014-08.org.nvmexpress:uuid:d3b581b4-c975-11e6-8425-0894ef31a074
```

5. On the ONTAP system, verify that the `hostnqn` string matches the `hostnqn` string for the corresponding subsystem on the ONTAP array:

```
::> vserver nvme subsystem host show -vserver vs_coexistence_emulex
```

Show example

```
Vserver Subsystem Priority  Host NQN
------- --------- --------
-------------------------------------------------
vs_coexistence_emulex
        nvme1
                  regular   nqn.2014-
08.org.nvmexpress:uuid:d3b581b4-c975-11e6-8425-0894ef31a074
        nvme10
                  regular   nqn.2014-
08.org.nvmexpress:uuid:d3b581b4-c975-11e6-8425-0894ef31a074
        nvme11
                  regular   nqn.2014-
08.org.nvmexpress:uuid:d3b581b4-c975-11e6-8425-0894ef31a074
        nvme12
                  regular   nqn.2014-
08.org.nvmexpress:uuid:d3b581b4-c975-11e6-8425-0894ef31a074
4 entries were displayed.
```

If the `hostnqn` strings do not match, use the `vserver modify` command to update the `hostnqn` string on your corresponding ONTAP array subsystem to match the `hostnqn` string from `/etc/nvme/hostnqn` on the host.

Step 3: Configure NVMe/FC and NVMe/TCP

Configure NVMe/FC with Broadcom/Emulex or Marvell/QLogic adapters, or configure NVMe/TCP using manual discovery and connect operations.

NVMe/FC - Broadcom/Emulex

Configure NVMe/FC for a Broadcom/Emulex FC adapter.

Steps

1. Verify that you are using the supported adapter model:
   a. Display the model names:

```
cat /sys/class/scsi_host/host*/modelname
```

   You should see the following output:

```
SN37A92079
SN37A92079
```

   b. Display the model descriptions:

```
cat /sys/class/scsi_host/host*/modeldesc
```

   You should see the following output:

```
Emulex SN37A92079 32Gb 2-Port Fibre Channel Adapter
Emulex SN37A92079 32Gb 2-Port Fibre Channel Adapter
```

2. Verify that you are using the recommended Broadcom `lpfc` firmware and inbox driver:
   a. Display the firmware version:

```
cat /sys/class/scsi_host/host*/fwrev
```

   The following example shows firmware versions:

```
14.4.393.53, sli-4:6:d
14.4.393.53, sli-4:6:d
```

   b. Display the inbox driver version:

```
cat /sys/module/lpfc/version
```

   The following example shows a driver version:

```
0:14.4.0.11
```

For the current list of supported adapter driver and firmware versions, see the Interoperability Matrix Tool.

3. Verify that the expected output of `lpfc_enable_fc4_type` is set to `3`:

```
cat /sys/module/lpfc/parameters/lpfc_enable_fc4_type
```

4. Verify that you can view your initiator ports:

```
cat /sys/class/fc_host/host*/port_name
```

You should see an output similar to:

```
0x100000109bdacc75
0x100000109bdacc76
```

5. Verify that your initiator ports are online:

```
cat /sys/class/fc_host/host*/port_state
```

You should see the following output:

```
Online
Online
```

6. Verify that the NVMe/FC initiator ports are enabled and that the target ports are visible:

```
cat /sys/class/scsi_host/host*/nvme_info
```

Show example output

```
NVME Initiator Enabled
XRI Dist lpfc0 Total 6144 IO 5894 ELS 250
NVME LPORT lpfc0 WWPN x100000109bdacc75 WWNN x200000109bdacc75
DID x060100 ONLINE
NVME RPORT       WWPN x2001d039ea951c45 WWNN x2000d039ea951c45
DID x080801 TARGET DISCSRVC ONLINE
NVME RPORT       WWPN x2003d039ea951c45 WWNN x2000d039ea951c45
DID x080d01 TARGET DISCSRVC ONLINE
NVME RPORT       WWPN x2024d039eab31e9c WWNN x2023d039eab31e9c
DID x020a09 TARGET DISCSRVC ONLINE
NVME RPORT       WWPN x2026d039eab31e9c WWNN x2023d039eab31e9c
DID x020a08 TARGET DISCSRVC ONLINE
NVME RPORT       WWPN x2003d039ea5cfc90 WWNN x2002d039ea5cfc90
DID x061b01 TARGET DISCSRVC ONLINE
NVME RPORT       WWPN x2012d039ea5cfc90 WWNN x2011d039ea5cfc90
DID x061b05 TARGET DISCSRVC ONLINE
NVME RPORT       WWPN x2005d039ea5cfc90 WWNN x2002d039ea5cfc90
DID x061201 TARGET DISCSRVC ONLINE
NVME RPORT       WWPN x2014d039ea5cfc90 WWNN x2011d039ea5cfc90
DID x061205 TARGET DISCSRVC ONLINE

NVME Statistics
LS: Xmt 0000017242 Cmpl 0000017242 Abort 00000000
LS XMIT: Err 00000000  CMPL: xb 00000000 Err 00000000
Total FCP Cmpl 0000000000378362 Issue 00000000003783c7 OutIO
0000000000000065
        abort 00000409 noxri 00000000 nondlp 0000003a qdepth
00000000 wqerr 00000000 err 00000000
FCP CMPL: xb 00000409 Err 0000040a

NVME Initiator Enabled
XRI Dist lpfc1 Total 6144 IO 5894 ELS 250
NVME LPORT lpfc1 WWPN x100000109bdacc76 WWNN x200000109bdacc76
DID x062800 ONLINE
NVME RPORT       WWPN x2002d039ea951c45 WWNN x2000d039ea951c45
DID x080701 TARGET DISCSRVC ONLINE
NVME RPORT       WWPN x2004d039ea951c45 WWNN x2000d039ea951c45
DID x081501 TARGET DISCSRVC ONLINE
NVME RPORT       WWPN x2025d039eab31e9c WWNN x2023d039eab31e9c
DID x020913 TARGET DISCSRVC ONLINE
NVME RPORT       WWPN x2027d039eab31e9c WWNN x2023d039eab31e9c
DID x020912 TARGET DISCSRVC ONLINE
NVME RPORT       WWPN x2006d039ea5cfc90 WWNN x2002d039ea5cfc90
DID x061401 TARGET DISCSRVC ONLINE
```

```
NVME RPORT       WWPN x2015d039ea5cfc90 WWNN x2011d039ea5cfc90
DID x061405 TARGET DISCSRVC ONLINE
NVME RPORT       WWPN x2004d039ea5cfc90 WWNN x2002d039ea5cfc90
DID x061301 TARGET DISCSRVC ONLINE
NVME RPORT       WWPN x2013d039ea5cfc90 WWNN x2011d039ea5cfc90
DID x061305 TARGET DISCSRVC ONLINE

NVME Statistics
LS: Xmt 0000017428 Cmpl 0000017428 Abort 00000000
LS XMIT: Err 00000000  CMPL: xb 00000000 Err 00000000
Total FCP Cmpl 00000000003443be Issue 000000000034442a OutIO
000000000000006c
        abort 00000491 noxri 00000000 nondlp 00000086 qdepth
00000000 wqerr 00000000 err 00000000
FCP CMPL: xb 00000491 Err 00000494
```

NVMe/FC - Marvell/QLogic

Configure NVMe/FC for a Marvell/QLogic adapter.

Steps

1. Verify that you are running the supported adapter driver and firmware versions:

   ```
   cat /sys/class/fc_host/host*/symbolic_name
   ```

   The follow example shows driver and firmware versions:

   ```
   QLE2772 FW:v9.15.06 DVR:v10.02.09.400-k-debug
   QLE2772 FW:v9.15.06 DVR:v10.02.09.400-k-debug
   ```

2. Verify that `ql2xnvmeenable` is set. This enables the Marvell adapter to function as an NVMe/FC initiator:

   ```
   cat /sys/module/qla2xxx/parameters/ql2xnvmeenable
   ```

   The expected output is 1.

NVMe/TCP

The NVMe/TCP protocol doesn't support the auto-connect operation. Instead, you can discover the NVMe/TCP subsystems and namespaces by performing the NVMe/TCP `connect` or `connect-all` operations manually.

Steps

1. Verify that the initiator port can fetch the discovery log page data across the supported NVMe/TCP

LIFs:

```
nvme discover -t tcp -w <host-traddr> -a <traddr>
```

Show example output

```
nvme discover -t tcp -w 192.168.38.20 -a 192.168.38.10
Discovery Log Number of Records 8, Generation counter 42
=====Discovery Log Entry 0======
trtype:  tcp
adrfam:  ipv4
subtype: current discovery subsystem
treq:    not specified
portid:  4
trsvcid: 8009
subnqn:  nqn.1992-
08.com.netapp:sn.f8e2af201b7211f0ac2bd039eab67a95:discovery
traddr:  192.168.211.71
eflags:  explicit discovery connections, duplicate discovery
information
sectype: none
=====Discovery Log Entry 1======
trtype:  tcp
adrfam:  ipv4
subtype: current discovery subsystem
treq:    not specified
portid:  3
trsvcid: 8009
subnqn:  nqn.1992-
08.com.netapp:sn.f8e2af201b7211f0ac2bd039eab67a95:discovery
traddr:  192.168.111.71
eflags:  explicit discovery connections, duplicate discovery
information
sectype: none
=====Discovery Log Entry 2======
trtype:  tcp
adrfam:  ipv4
subtype: current discovery subsystem
treq:    not specified
portid:  2
trsvcid: 8009
subnqn:  nqn.1992-
08.com.netapp:sn.f8e2af201b7211f0ac2bd039eab67a95:discovery
traddr:  192.168.211.70
eflags:  explicit discovery connections, duplicate discovery
information
sectype: none
=====Discovery Log Entry 3======
trtype:  tcp
adrfam:  ipv4
```

```
subtype: current discovery subsystem
treq:    not specified
portid:  1
trsvcid: 8009
subnqn:  nqn.1992-
08.com.netapp:sn.f8e2af201b7211f0ac2bd039eab67a95:discovery
traddr:  192.168.111.70
eflags:  explicit discovery connections, duplicate discovery
information
sectype: none
=====Discovery Log Entry 4======
trtype:  tcp
adrfam:  ipv4
subtype: nvme subsystem
treq:    not specified
portid:  4
trsvcid: 4420
subnqn:  nqn.1992-
08.com.netapp:sn.f8e2af201b7211f0ac2bd039eab67a95:subsystem.samp
le_tcp_sub
traddr:  192.168.211.71
eflags:  none
sectype: none
=====Discovery Log Entry 5======
trtype:  tcp
adrfam:  ipv4
subtype: nvme subsystem
treq:    not specified
portid:  3
trsvcid: 4420
subnqn:  nqn.1992-
08.com.netapp:sn.f8e2af201b7211f0ac2bd039eab67a95:subsystem.samp
le_tcp_sub
traddr:  192.168.111.71
eflags:  none
sectype: none
=====Discovery Log Entry 6======
trtype:  tcp
adrfam:  ipv4
subtype: nvme subsystem
treq:    not specified
portid:  2
trsvcid: 4420
subnqn:  nqn.1992-
08.com.netapp:sn.f8e2af201b7211f0ac2bd039eab67a95:subsystem.samp
le_tcp_sub
```

```
traddr:  192.168.211.70
eflags:  none
sectype: none
=====Discovery Log Entry 7======
trtype:  tcp
adrfam:  ipv4
subtype: nvme subsystem
treq:    not specified
portid:  1
trsvcid: 4420
subnqn:  nqn.1992-
08.com.netapp:sn.f8e2af201b7211f0ac2bd039eab67a95:subsystem.samp
le_tcp_sub
traddr:  192.168.111.70
eflags:  none
sectype: none
localhost:~ #
```

2. Verify that all other NVMe/TCP initiator-target LIF combinations can successfully fetch discovery log page data:

```
nvme discover -t tcp -w <host-traddr> -a <traddr>
```

Show example

```
nvme discover -t tcp -w 192.168.38.20 -a 192.168.38.10
nvme discover -t tcp -w 192.168.38.20 -a 192.168.38.11
nvme discover -t tcp -w 192.168.39.20 -a 192.168.39.10
nvme discover -t tcp -w 192.168.39.20 -a 192.168.39.11
```

3. Run the `nvme connect-all` command across all the supported NVMe/TCP initiator-target LIFs across the nodes:

```
nvme connect-all -t tcp -w <host-traddr> -a <traddr>
```

Show example

```
nvme    connect-all -t  tcp -w  192.168.38.20   -a
192.168.38.10
nvme    connect-all -t  tcp -w  192.168.38.20   -a
192.168.38.11
nvme    connect-all -t  tcp -w  192.168.39.20   -a
192.168.39.10
nvme    connect-all -t  tcp -w  192.168.39.20   -a
192.168.39.11
```

The setting for the NVMe/TCP `ctrl_loss_tmo timeout` is automatically set to "off". As a result:

- There are no limits on the number of retries (indefinite retry).
- You don't need to manually configure a specific `ctrl_loss_tmo timeout` duration when using the `nvme connect` or `nvme connect-all` commands (option -l).
- The NVMe/TCP controllers don't experience timeouts in the event of a path failure and remain connected indefinitely.

Step 4: Optionally, modify the iopolicy in the udev rules

Beginning with SUSE Linux Enterprise Server 16, the default iopolicy for NVMe-oF is set to `queue-depth`. If you want to change the iopolicy to `round-robin`, modify the udev rules file as follows:

Steps

1. Open the udev rules file in a text editor with root privileges:

   ```
   /usr/lib/udev/rules.d/71-nvmf-netapp.rules
   ```

   You should see the following output:

   ```
   vi /usr/lib/udev/rules.d/71-nvmf-netapp.rules
   ```

2. Find the line that sets iopolicy for the NetApp ONTAP Controller, as shown in the following example rule:

   ```
   ACTION=="add", SUBSYSTEM=="nvme-subsystem", ATTR{subsystype}=="nvm",
   ATTR{model}=="NetApp ONTAP Controller", ATTR{iopolicy}="queue-depth"
   ```

3. Modify the rule so that `queue-depth` becomes `round-robin`:

```
ACTION=="add", SUBSYSTEM=="nvme-subsystem", ATTR{subsystype}=="nvm",
ATTR{model}=="NetApp ONTAP Controller", ATTR{iopolicy}="round-robin"
```

4. Reload the udev rules and apply the changes:

```
udevadm control --reload
udevadm trigger --subsystem-match=nvme-subsystem
```

5. Verify the current iopolicy for your subsystem. Replace <subsystem>, for example, `nvme-subsys0`.

```
cat /sys/class/nvme-subsystem/<subsystem>/iopolicy
```

You should see the following output:

```
round-robin
```

The new iopolicy applies automatically to matching NetApp ONTAP Controller devices. You don't need to reboot.

Step 5: Optionally, enable 1MB I/O for NVMe/FC

ONTAP reports a Max Data Transfer Size (MDTS) of 8 in the Identify Controller data. This means the maximum I/O request size can be up to 1MB. To issue I/O requests of size 1MB for a Broadcom NVMe/FC host, you should increase the `lpfc` value of the `lpfc_sg_seg_cnt` parameter to 256 from the default value of 64.

These steps don't apply to Qlogic NVMe/FC hosts.

Steps

1. Set the `lpfc_sg_seg_cnt` parameter to 256:

```
cat /etc/modprobe.d/lpfc.conf
```

You should see an output similar to the following example:

```
options lpfc lpfc_sg_seg_cnt=256
```

2. Run the `dracut -f` command, and reboot the host.
3. Verify that the value for `lpfc_sg_seg_cnt` is 256:

```
cat /sys/module/lpfc/parameters/lpfc_sg_seg_cnt
```

Step 6: Verify NVMe boot services

The `nvmefc-boot-connections.service` and `nvmf-autoconnect.service` boot services included in the NVMe/FC `nvme-cli` package are automatically enabled when the system boots.

After booting completes, verify that the `nvmefc-boot-connections.service` and `nvmf-autoconnect.service` boot services are enabled.

Steps

1. Verify that `nvmf-autoconnect.service` is enabled:

```
systemctl status nvmf-autoconnect.service
```

Show example output

```
nvmf-autoconnect.service - Connect NVMe-oF subsystems automatically
during boot
  Loaded: loaded (/usr/lib/systemd/system/nvmf-autoconnect.service;
enabled; vendor preset: disabled)
  Active: inactive (dead) since Thu 2024-05-25 14:55:00 IST; 11min
ago
Process: 2108 ExecStartPre=/sbin/modprobe nvme-fabrics (code=exited,
status=0/SUCCESS)
Process: 2114 ExecStart=/usr/sbin/nvme connect-all (code=exited,
status=0/SUCCESS)
Main PID: 2114 (code=exited, status=0/SUCCESS)

systemd[1]: Starting Connect NVMe-oF subsystems automatically during
boot...
nvme[2114]: traddr=nn-0x201700a098fd4ca6:pn-0x201800a098fd4ca6 is
already connected
systemd[1]: nvmf-autoconnect.service: Deactivated successfully.
systemd[1]: Finished Connect NVMe-oF subsystems automatically during
boot.
```

2. Verify that `nvmefc-boot-connections.service` is enabled:

```
systemctl status nvmefc-boot-connections.service
```

Show example output

```
nvmefc-boot-connections.service - Auto-connect to subsystems on FC-
NVME devices found during boot
   Loaded: loaded (/usr/lib/systemd/system/nvmefc-boot-
connections.service; enabled; vendor preset: enabled)
   Active: inactive (dead) since Thu 2024-05-25 14:55:00 IST; 11min
ago
 Main PID: 1647 (code=exited, status=0/SUCCESS)

systemd[1]: Starting Auto-connect to subsystems on FC-NVME devices
found during boot...
systemd[1]: nvmefc-boot-connections.service: Succeeded.
systemd[1]: Finished Auto-connect to subsystems on FC-NVME devices
found during boot.
```

Step 7: Verify the multipathing configuration

Verify that the in-kernel NVMe multipath status, ANA status, and ONTAP namespaces are correct for the NVMe-oF configuration.

Steps

1. Verify that the in-kernel NVMe multipath is enabled:

   ```
   cat /sys/module/nvme_core/parameters/multipath
   ```

   You should see the following output:

   ```
   Y
   ```

2. Verify that the appropriate NVMe-oF settings (such as, model set to NetApp ONTAP Controller and load balancing iopolicy set to queue-depth) for the respective ONTAP namespaces correctly reflect on the host:

   a. Display the subsystems:

      ```
      cat /sys/class/nvme-subsystem/nvme-subsys*/model
      ```

      You should see the following output:

      ```
      NetApp ONTAP Controller
      NetApp ONTAP Controller
      ```

b. Display the policy:

```
cat /sys/class/nvme-subsystem/nvme-subsys*/iopolicy
```

You should see the following output:

```
queue-depth
queue-depth
```

3. Verify that the namespaces are created and correctly discovered on the host:

```
nvme list
```

Show example

```
Node         SN                   Model
------------------------------------------------------------
/dev/nvme7n1 81Ix2BVuekWcAAAAAAAB   NetApp ONTAP Controller

Namespace Usage    Format             FW             Rev
-----             21.47 GB / 21.47 GB    4 KiB + 0 B   FFFFFFFF
```

4. Verify that the controller state of each path is live and has the correct ANA status:

```
nvme list-subsys /dev/<controller_ID>
```

Beginning with ONTAP 9.16.1, NVMe/FC and NVMe/TCP report all optimized paths on ASA r2 systems.

NVMe/FC

This following example outputs show a namespace hosted on a two-node ONTAP controller for AFF, FAS, and ASA systems and ASA r2 system with NVMe/FC.

Show AFF, FAS, and ASA example output

```
 nvme-subsys114 - NQN=nqn.1992-
08.com.netapp:sn.9e30b9760a4911f08c87d039eab67a95:subsystem.sles
_161_27
                 hostnqn=nqn.2014-
08.org.nvmexpress:uuid:f6517cae-3133-11e8-bbff-7ed30aef123f
iopolicy=round-robin\
+- nvme114 fc traddr=nn-0x234ed039ea359e4a:pn-
0x2360d039ea359e4a,host_traddr=nn-0x20000090fae0ec88:pn-
0x10000090fae0ec88 live optimized
+- nvme115 fc traddr=nn-0x234ed039ea359e4a:pn-
0x2362d039ea359e4a,host_traddr=nn-0x20000090fae0ec88:pn-
0x10000090fae0ec88 live non-optimized
+- nvme116 fc traddr=nn-0x234ed039ea359e4a:pn-
0x2361d039ea359e4a,host_traddr=nn-0x20000090fae0ec89:pn-
0x10000090fae0ec89 live optimized
+- nvme117 fc traddr=nn-0x234ed039ea359e4a:pn-
0x2363d039ea359e4a,host_traddr=nn-0x20000090fae0ec89:pn-
0x10000090fae0ec89 live non-optimized
```

Show ASA r2 example output

```
nvme-subsys96 - NQN=nqn.1992-
08.om.netapp:sn.b351b2b6777b11f0b3c2d039ea5cfc91:subsystem.nvme2
4
                hostnqn=nqn.2014-
08.org.nvmexpress:uuid:d3b581b4-c975-11e6-8425-0894ef31a074
\
 +- nvme203 fc traddr=nn-0x2011d039ea5cfc90:pn-
0x2015d039ea5cfc90,host_traddr=nn-0x200000109bdacc76:pn-
0x100000109bdacc76 live optimized
 +- nvme25 fc traddr=nn-0x2011d039ea5cfc90:pn-
0x2014d039ea5cfc90,host_traddr=nn-0x200000109bdacc75:pn-
0x100000109bdacc75 live optimized
 +- nvme30 fc traddr=nn-0x2011d039ea5cfc90:pn-
0x2012d039ea5cfc90,host_traddr=nn-0x200000109bdacc75:pn-
0x100000109bdacc75 live optimized
 +- nvme32 fc traddr=nn-0x2011d039ea5cfc90:pn-
0x2013d039ea5cfc90,host_traddr=nn-0x200000109bdacc76:pn-
0x100000109bdacc76 live optimized
```

NVMe/TCP

This following example outputs show a namespace hosted on a two-node ONTAP controller for AFF, FAS, and ASA systems and ASA r2 systems with NVMe/TCP.

Show AFF, FAS, and ASA example output

```
nvme-subsys9 - NQN=nqn.1992-
08.com.netapp:sn.9927e165694211f0b4f4d039eab31e9d:subsystem.nvme
10
               hostnqn=nqn.2014-08.org.nvmexpress:uuid:4c4c4544-
0035-5910-804b-b7c04f444d33
\
 +- nvme105 tcp
traddr=192.168.39.10,trsvcid=4420,host_traddr=192.168.39.20,src_
addr=192.168.39.20 live optimized
 +- nvme153 tcp
traddr=192.168.39.11,trsvcid=4420,host_traddr=192.168.39.20,src_
addr=192.168.39.20 live non-optimized
 +- nvme57 tcp
traddr=192.168.38.11,trsvcid=4420,host_traddr=192.168.38.20,src_
addr=192.168.38.20 live non-optimized
 +- nvme9 tcp
traddr=192.168.38.10,trsvcid=4420,host_traddr=192.168.38.20,src_
addr=192.168.38.20 live optimized
```

Show ASA r2 example output

```
nvme-subsys4 - NQN=nqn.1992-
08.com.netapp:sn.17e32b6e8c7f11f09545d039eac03c33:subsystem.Bidi
rectional_DHCP_1_0
               hostnqn=nqn.2014-08.org.nvmexpress:uuid:4c4c4544-
0054-5110-8039-c3c04f523034
\
+- nvme4 tcp
traddr=192.168.20.28,trsvcid=4420,host_traddr=192.168.20.21,src_
addr=192.168.20.21 live optimized
+- nvme5 tcp
traddr=192.168.20.29,trsvcid=4420,host_traddr=192.168.20.21,src_
addr=192.168.20.21 live optimized
+- nvme6 tcp
traddr=192.168.21.28,trsvcid=4420,host_traddr=192.168.21.21,src_
addr=192.168.21.21 live optimized
+- nvme7 tcp
traddr=192.168.21.29,trsvcid=4420,host_traddr=192.168.21.21,src_
addr=192.168.21.21 live optimized
```

5. Verify that the NetApp plug-in displays the correct values for each ONTAP namespace device:

Column

```
nvme netapp ontapdevices -o column
```

Show example

```
Device           Vserver                   Namespace Path
NSID UUID                                   Size
---------------- ------------------------- ------------------
---- ---------------------------------------- ---------
/dev/nvme0n1     vs_coexistence_emulex     ns1                1
79510f05-7784-11f0-b3c2-d039ea5cfc91   21.47GB
```

JSON

```
nvme netapp ontapdevices -o json
```

Show example

```
{
"ONTAPdevices":[{
      "Device":"/dev/nvme0n1",
      "Vserver":"vs_coexistence_emulex",
      "Namespace_Path":"ns1",
      "NSID":1,
      "UUID":"79510f05-7784-11f0-b3c2-d039ea5cfc91",
      "Size":"21.47GB",
      "LBA_Data_Size":4096,
      "Namespace_Size":5242880
    }  ]
}
```

Step 8: Create a persistent discovery controller

You can create a persistent discovery controller (PDC) for a SUSE Linux Enterprise Server 16 host. A PDC is required to automatically detect an NVMe subsystem add or remove operation and changes to the discovery log page data.

Steps

1. Verify that the discovery log page data is available and can be retrieved through the initiator port and target LIF combination:

```
nvme discover -t <trtype> -w <host-traddr> -a <traddr>
```

Show example output

```
Discovery Log Number of Records 8, Generation counter 10
=====Discovery Log Entry 0======
trtype:  tcp
adrfam:  ipv4
subtype: current discovery subsystem
treq:    not specified
portid:  3
trsvcid: 8009
subnqn:  nqn.1992-
08.com.netapp:sn.9927e165694211f0b4f4d039eab31e9d:discovery
traddr:  192.168.39.10
eflags:  explicit discovery connections, duplicate discovery
information
sectype: none
=====Discovery Log Entry 1======
trtype:  tcp
adrfam:  ipv4
subtype: current discovery subsystem
treq:    not specified
portid:  1
trsvcid: 8009
subnqn:  nqn.1992-
08.com.netapp:sn.9927e165694211f0b4f4d039eab31e9d:discovery
traddr:  192.168.38.10
eflags:  explicit discovery connections, duplicate discovery
information
sectype: none
=====Discovery Log Entry 2======
trtype:  tcp
adrfam:  ipv4
subtype: current discovery subsystem
treq:    not specified
portid:  4
trsvcid: 8009
subnqn:  nqn.1992-
08.com.netapp:sn.9927e165694211f0b4f4d039eab31e9d:discovery
traddr:  192.168.39.11
eflags:  explicit discovery connections, duplicate discovery
information
sectype: none
=====Discovery Log Entry 3======
trtype:  tcp
adrfam:  ipv4
subtype: current discovery subsystem
```

```
treq:    not specified
portid:  2
trsvcid: 8009
subnqn:  nqn.1992-
08.com.netapp:sn.9927e165694211f0b4f4d039eab31e9d:discovery
traddr:  192.168.38.11
eflags:  explicit discovery connections, duplicate discovery
information
sectype: none
=====Discovery Log Entry 4======
trtype:  tcp
adrfam:  ipv4
subtype: nvme subsystem
treq:    not specified
portid:  3
trsvcid: 4420
subnqn:  nqn.1992-
08.com.netapp:sn.9927e165694211f0b4f4d039eab31e9d:subsystem.nvme1
traddr:  192.168.39.10
eflags:  none
sectype: none
=====Discovery Log Entry 5======
trtype:  tcp
adrfam:  ipv4
subtype: nvme subsystem
treq:    not specified
portid:  1
trsvcid: 4420
subnqn:  nqn.1992-
08.com.netapp:sn.9927e165694211f0b4f4d039eab31e9d:subsystem.nvme1
traddr:  192.168.38.10
eflags:  none
sectype: none
=====Discovery Log Entry 6======
trtype:  tcp
adrfam:  ipv4
subtype: nvme subsystem
treq:    not specified
portid:  4
trsvcid: 4420
subnqn:  nqn.1992-
08.com.netapp:sn.9927e165694211f0b4f4d039eab31e9d:subsystem.nvme1
traddr:  192.168.39.11
eflags:  none
sectype: none
=====Discovery Log Entry 7======
```

```
trtype:  tcp
adrfam:  ipv4
subtype: nvme subsystem
treq:    not specified
portid:  2
trsvcid: 4420
subnqn:  nqn.1992-
08.com.netapp:sn.9927e165694211f0b4f4d039eab31e9d:subsystem.nvme1
traddr:  192.168.38.11
eflags:  none
sectype: none
```

2. Create a PDC for the discovery subsystem:

```
nvme discover -t <trtype> -w <host-traddr> -a <traddr> -p
```

You should see the following output:

```
nvme discover -t tcp -w 192.168.39.20 -a 192.168.39.11 -p
```

3. From the ONTAP controller, verify that the PDC has been created:

```
vserver nvme show-discovery-controller -instance -vserver <vserver_name>
```

Show example output

```
vserver nvme show-discovery-controller -instance -vserver
vs_tcp_sles16
Vserver Name: vs_tcp_sles16
               Controller ID: 0180h
     Discovery Subsystem NQN: nqn.1992-
08.com.netapp:sn.9927e165694211f0b4f4d039eab31e9d:discovery
           Logical Interface: lif3
                        Node: A400-12-171
                    Host NQN: nqn.2014-
08.org.nvmexpress:uuid:4c4c4544-0035-5910-804b-b7c04f444d33
          Transport Protocol: nvme-tcp
 Initiator Transport Address: 192.168.39.20
Transport Service Identifier: 8009
             Host Identifier: 4c4c454400355910804bb7c04f444d33
            Admin Queue Depth: 32
       Header Digest Enabled: false
         Data Digest Enabled: false
   Keep-Alive Timeout (msec): 30000
```

Step 9: Set up secure in-band authentication

Secure in-band authentication is supported over NVMe/TCP between a SUSE Linux Enterprise Server 16 host and an ONTAP controller.

Each host or controller must be associated with a `DH-HMAC-CHAP` key to set up secure authentication. A DH-HMAC-CHAP key is a combination of the NQN of the NVMe host or controller and an authentication secret configured by the administrator. To authenticate its peer, an NVMe host or controller must recognize the key associated with the peer.

Steps

Set up secure in-band authentication using the CLI or a config JSON file. If you need to specify different dhchap keys for different subsystems, you must use a config JSON file.

CLI

Set up secure in-band authentication using the CLI.

1. Obtain the host NQN:

```
cat /etc/nvme/hostnqn
```

2. Generate the dhchap key for the host.

   The following output describes the `gen-dhchap-key` command paramters:

```
nvme gen-dhchap-key -s optional_secret -l key_length {32|48|64} -m
HMAC_function {0|1|2|3} -n host_nqn
•   -s secret key in hexadecimal characters to be used to initialize
the host key
•   -l length of the resulting key in bytes
•   -m HMAC function to use for key transformation
0 = none, 1- SHA-256, 2 = SHA-384, 3=SHA-512
•   -n host NQN to use for key transformation
```

   In the following example, a random dhchap key with HMAC set to 3 (SHA-512) is generated.

```
nvme gen-dhchap-key -m 3 -n nqn.2014-
08.org.nvmexpress:uuid:4c4c4544-0035-5910-804b-b7c04f444d33
DHHC-
1:03:ohdxI1yIS8gBLwIOubcwl57rXcozYuRgBsoWaBvxEvpDlQHn/7dQ4JjFGwmhgwd
JWmVoripbWbMJy5eMAbCahN4hhYU=:
```

3. On the ONTAP controller, add the host and specify both dhchap keys:

```
vserver nvme subsystem host add -vserver <svm_name> -subsystem
<subsystem> -host-nqn <host_nqn> -dhchap-host-secret
<authentication_host_secret> -dhchap-controller-secret
<authentication_controller_secret> -dhchap-hash-function {sha-
256|sha-512} -dhchap-group {none|2048-bit|3072-bit|4096-bit|6144-
bit|8192-bit}
```

4. A host supports two types of authentication methods, unidirectional and bidirectional. On the host, connect to the ONTAP controller and specify dhchap keys based on the chosen authentication method:

```
nvme connect -t tcp -w <host-traddr> -a <tr-addr> -n <host_nqn> -S
<authentication_host_secret> -C <authentication_controller_secret>
```

5. Validate the `nvme connect authentication` command by verifying the host and controller dhchap keys:

   a. Verify the host dhchap keys:

   ```
   cat /sys/class/nvme-subsystem/<nvme-subsysX>/nvme*/dhchap_secret
   ```

   Show example output for a unidirectional configuration

   ```
   # cat /sys/class/nvme-subsystem/nvme-
   subsys1/nvme*/dhchap_secret
   DHHC-1:01:wkwAKk8r9Ip7qECKt7V5aIo/7Y1CH7DWkUfLfMxmseg39DFb:
   DHHC-1:01:wkwAKk8r9Ip7qECKt7V5aIo/7Y1CH7DWkUfLfMxmseg39DFb:
   DHHC-1:01:wkwAKk8r9Ip7qECKt7V5aIo/7Y1CH7DWkUfLfMxmseg39DFb:
   DHHC-1:01:wkwAKk8r9Ip7qECKt7V5aIo/7Y1CH7DWkUfLfMxmseg39DFb:
   ```

   b. Verify the controller dhchap keys:

   ```
   cat /sys/class/nvme-subsystem/<nvme-
   subsysX>/nvme*/dhchap_ctrl_secret
   ```

   Show example output for a bidirectional configuration

   ```
   # cat /sys/class/nvme-subsystem/nvme-
   subsys6/nvme*/dhchap_ctrl_secret
   DHHC-
   1:03:ohdxI1yIS8gBLwIOubcwl57rXcozYuRgBsoWaBvxEvpDlQHn/7dQ4JjFG
   wmhgwdJWmVoripbWbMJy5eMAbCahN4hhYU=:
   DHHC-
   1:03:ohdxI1yIS8gBLwIOubcwl57rXcozYuRgBsoWaBvxEvpDlQHn/7dQ4JjFG
   wmhgwdJWmVoripbWbMJy5eMAbCahN4hhYU=:
   DHHC-
   1:03:ohdxI1yIS8gBLwIOubcwl57rXcozYuRgBsoWaBvxEvpDlQHn/7dQ4JjFG
   wmhgwdJWmVoripbWbMJy5eMAbCahN4hhYU=:
   DHHC-
   1:03:ohdxI1yIS8gBLwIOubcwl57rXcozYuRgBsoWaBvxEvpDlQHn/7dQ4JjFG
   wmhgwdJWmVoripbWbMJy5eMAbCahN4hhYU=:
   ```

JSON

When multiple NVMe subsystems are available on the ONTAP controller configuration, you can use the `/etc/nvme/config.json` file with the `nvme connect-all` command.

Use the `-o` option to generate the JSON file. Refer to the NVMe connect-all man pages for more syntax options.

1. Configure the JSON file:

Show example output

```
# cat /etc/nvme/config.json
[
  {
    "hostnqn":"nqn.2014-08.org.nvmexpress:uuid:4c4c4544-0035-
5910-804b-b7c04f444d33",
    "hostid":"4c4c4544-0035-5910-804b-b7c04f444d33",
    "dhchap_key":"DHHC-
1:01:wkwAKk8r9Ip7qECKt7V5aIo/7Y1CH7DWkUfLfMxmseg39DFb:",
    "subsystems":[
      {
        "nqn":"nqn.1992-
08.com.netapp:sn.9927e165694211f0b4f4d039eab31e9d:subsystem.inba
nd_bidirectional",
        "ports":[
          {
            "transport":"tcp",
            "traddr":"192.168.38.10",
            "host_traddr":"192.168.38.20",
            "trsvcid":"4420",
            "dhchap_ctrl_key":"DHHC-
1:03:ohdxI1yIS8gBLwIOubcwl57rXcozYuRgBsoWaBvxEvpDlQHn/7dQ4JjFGwm
hgwdJWmVoripbWbMJy5eMAbCahN4hhYU=:"
          },
          {
            "transport":"tcp",
            "traddr":"192.168.38.11",
            "host_traddr":"192.168.38.20",
            "trsvcid":"4420",
            "dhchap_ctrl_key":"DHHC-
1:03:ohdxI1yIS8gBLwIOubcwl57rXcozYuRgBsoWaBvxEvpDlQHn/7dQ4JjFGwm
hgwdJWmVoripbWbMJy5eMAbCahN4hhYU=:"
          },
          {
            "transport":"tcp",
            "traddr":"192.168.39.11",
            "host_traddr":"192.168.39.20",
            "trsvcid":"4420",
            "dhchap_ctrl_key":"DHHC-
1:03:ohdxI1yIS8gBLwIOubcwl57rXcozYuRgBsoWaBvxEvpDlQHn/7dQ4JjFGwm
hgwdJWmVoripbWbMJy5eMAbCahN4hhYU=:"
          },
          {
            "transport":"tcp",
            "traddr":"192.168.39.10",
```

```
            "host_traddr":"192.168.39.20",
            "trsvcid":"4420",
            "dhchap_ctrl_key":"DHHC-
1:03:ohdxI1yIS8gBLwIOubcwl57rXcozYuRgBsoWaBvxEvpDlQHn/7dQ4JjFGwm
hgwdJWmVoripbWbMJy5eMAbCahN4hhYU=:"
          }
        ]
      }
    ]
  }
]
```

In the following example, `dhchap_key` corresponds to `dhchap_secret` and `dhchap_ctrl_key` corresponds to `dhchap_ctrl_secret`.

2. Connect to the ONTAP controller using the config JSON file:

```
nvme connect-all -J /etc/nvme/config.json
```

Show example output

```
traddr=192.168.38.10is already connected
traddr=192.168.39.10 is already connected
traddr=192.168.38.11 is already connected
traddr=192.168.39.11 is already connected
traddr=192.168.38.10is already connected
traddr=192.168.39.10 is already connected
traddr=192.168.38.11 is already connected
traddr=192.168.39.11 is already connected
traddr=192.168.38.10is already connected
traddr=192.168.39.10 is already connected
traddr=192.168.38.11 is already connected
traddr=192.168.39.11 is already connected
```

3. Verify that the dhchap secrets have been enabled for the respective controllers for each subsystem:
   a. Verify the host dhchap keys:

```
cat /sys/class/nvme-subsystem/nvme-subsys0/nvme0/dhchap_secret
```

   The following example shows a dhchap key:

```
DHHC-1:01:wkwAKk8r9Ip7qECKt7V5aIo/7Y1CH7DWkUfLfMxmseg39DFb:
```

b. Verify the controller dhchap keys:

```
cat /sys/class/nvme-subsystem/nvme-
subsys0/nvme0/dhchap_ctrl_secret
```

You should see an output similar to the following example:

```
DHHC-
1:03:ohdxI1yIS8gBLwIOubcwl57rXcozYuRgBsoWaBvxEvpDlQHn/7dQ4JjFGwmh
gwdJWmVoripbWbMJy5eMAbCahN4hhYU=:
```

Step 10: Configure Transport Layer Security

Transport Layer Security (TLS) provides secure end-to-end encryption for NVMe connections between NVMe-oF hosts and an ONTAP array. You can configure TLS 1.3 using the CLI and a configured pre-shared key (PSK).

Perform the following steps on the SUSE Linux Enterprise Server host, except where it specifies that you perform a step on the ONTAP controller.

Steps

1. Check that you have the following `ktls-utils`, `openssl`, and `libopenssl` packages installed on the host:

   a. Verify the `ktls-utils`:

   ```
   rpm -qa | grep ktls
   ```

   You should see the following output displayed:

   ```
   ktls-utils-0.10+33.g311d943-160000.2.2.x86_64
   ```

   b. Verify the SSL packages:

   ```
   rpm -qa | grep ssl
   ```

Show example output

```
libopenssl3-3.5.0-160000.3.2.x86_64
openssl-3.5.0-160000.2.2.noarch
openssl-3-3.5.0-160000.3.2.x86_64
libopenssl3-x86-64-v3-3.5.0-160000.3.2.x86_64
```

2. Verify that you have the correct setup for `/etc/tlshd.conf`:

```
cat /etc/tlshd.conf
```

Show example output

```
[debug]
loglevel=0
tls=0
nl=0

[authenticate]
#keyrings= <keyring>;<keyring>;<keyring>

[authenticate.client]
#x509.truststore= <pathname>
#x509.certificate= <pathname>
#x509.private_key= <pathname>

[authenticate.server]
#x509.truststore= <pathname>
#x509.certificate= <pathname>
#x509.private_key= <pathname>
```

3. Enable `tlshd` to start at system boot:

```
systemctl enable tlshd
```

4. Verify that the `tlshd` daemon is running:

```
systemctl status tlshd
```

Show example output

```
tlshd.service - Handshake service for kernel TLS consumers
   Loaded: loaded (/usr/lib/systemd/system/tlshd.service; enabled;
preset: disabled)
   Active: active (running) since Wed 2024-08-21 15:46:53 IST; 4h
57min ago
     Docs: man:tlshd(8)
Main PID: 961 (tlshd)
   Tasks: 1
     CPU: 46ms
   CGroup: /system.slice/tlshd.service
       └─961 /usr/sbin/tlshd
Aug 21 15:46:54 RX2530-M4-17-153 tlshd[961]: Built from ktls-utils
0.11-dev on Mar 21 2024 12:00:00
```

5. Generate the TLS PSK by using the `nvme gen-tls-key`:
   a. Verify the host:

      ```
      cat /etc/nvme/hostnqn
      ```

      You should see the following output:

      ```
      nqn.2014-08.org.nvmexpress:uuid:4c4c4544-0035-5910-804b-b7c04f444d33
      ```

   b. Verify the key:

      ```
      nvme gen-tls-key --hmac=1 --identity=1 --subsysnqn= nqn.1992-
      08.com.netapp:sn.9927e165694211f0b4f4d039eab31e9d:subsystem.nvme1
      ```

      You should see the following output:

      ```
      NVMeTLSkey-1:01:C50EsaGtuOp8n5fGE9EuWjbBCtshmfoHx4XTqTJUmydf0gIj:
      ```

6. On the ONTAP controller, add the TLS PSK to the ONTAP subsystem:

Show example output

```
nvme subsystem host add -vserver vs_iscsi_tcp -subsystem nvme1 -host
-nqn nqn.2014-08.org.nvmexpress:uuid:4c4c4544-0035-5910-804b-
b2c04f444d33 -tls-configured-psk NVMeTLSkey-
1:01:C50EsaGtuOp8n5fGE9EuWjbBCtshmfoHx4XTqTJUmydf0gIj:
```

7. Insert the TLS PSK into the host kernel keyring:

```
nvme check-tls-key --identity=1 --subsysnqn=nqn.1992
-08.com.netapp:sn.9927e165694211f0b4f4d039eab31e9d:subsystem.nvme1
--keydata=NVMeTLSkey
-1:01:C50EsaGtuOp8n5fGE9EuWjbBCtshmfoHx4XTqTJUmydf0gIj: --insert
```

You should see the following TLS key:

```
Inserted TLS key 069f56bb
```

The PSK shows as `NVMe1R01` because it uses `identity v1` from the TLS handshake algorithm. Identity v1 is the only version that ONTAP supports.

8. Verify that the TLS PSK is inserted correctly:

```
cat /proc/keys | grep NVMe
```

Show example output

```
069f56bb I-Q-- 5 perm 3b010000 0 0 psk NVMe1R01 nqn.2014-
08.org.nvmexpress:uuid:4c4c4544-0035-5910-804b-b2c04f444d33
nqn.1992-
08.com.netapp:sn.9927e165694211f0b4f4d039eab31e9d:subsystem.nvme1
oYVLelmiOwnvDjXKBmrnIgGVpFIBDJtc4hmQXE/36Sw=: 32
```

9. Connect to the ONTAP subsystem using the inserted TLS PSK:
   a. Verify the TLS PSK:

```
nvme connect -t tcp -w 192.168.38.20 -a 192.168.38.10  -n nqn.1992-
08.com.netapp:sn.9927e165694211f0b4f4d039eab31e9d:subsystem.nvme1
--tls_key=0x069f56bb –tls
```

You should see the following output:

```
connecting to device: nvme0
```

b. Verify the list-subsys:

```
nvme list-subsys
```

Show example output

```
nvme-subsys0 - NQN=nqn.1992-
08.com.netapp:sn.9927e165694211f0b4f4d039eab31e9d:subsystem.nvme1
               hostnqn=nqn.2014-08.org.nvmexpress:uuid:4c4c4544-
0035-5910-804b-b2c04f444d33
**
 +- nvme0 tcp
traddr=192.168.38.10,trsvcid=4420,host_traddr=192.168.38.20,src_a
ddr=192.168.38.20 live
```

10. Add the target, and verify the TLS connection to the specified ONTAP subsystem:

```
nvme subsystem controller show -vserver vs_tcp_sles16 -subsystem nvme1
-instance
```

Show example output

```
(vserver nvme subsystem controller show)
                      Vserver Name: vs_tcp_sles16
                         Subsystem: nvme1
                     Controller ID: 0040h
                  Logical Interface: lif1
                              Node: A400-12-171
                          Host NQN: nqn.2014-
08.org.nvmexpress:uuid:4c4c4544-0035-5910-804b-b2c04f444d33
                 Transport Protocol: nvme-tcp
        Initiator Transport Address: 192.168.38.20
                    Host Identifier:
4c4c454400355910804bb2c04f444d33
               Number of I/O Queues: 2
                   I/O Queue Depths: 128, 128
                  Admin Queue Depth: 32
              Max I/O Size in Bytes: 1048576
          Keep-Alive Timeout (msec): 5000
                     Subsystem UUID: 62203cfd-826a-11f0-966e-
d039eab31e9d
              Header Digest Enabled: false
                Data Digest Enabled: false
       Authentication Hash Function: sha-256
Authentication Diffie-Hellman Group: 3072-bit
                Authentication Mode: unidirectional
       Transport Service Identifier: 4420
                       TLS Key Type: configured
                   TLS PSK Identity: NVMe1R01 nqn.2014-
08.org.nvmexpress:uuid:4c4c4544-0035-5910-804b-b2c04f444d33
nqn.1992-
08.com.netapp:sn.9927e165694211f0b4f4d039eab31e9d:subsystem.nvme1
oYVLelmiOwnvDjXKBmrnIgGVpFIBDJtc4hmQXE/36Sw=
                         TLS Cipher: TLS-AES-128-GCM-SHA256
```

Step 11: Review the known issues

There are no known issues.

Configure SUSE Linux Enterprise Server 15 SPx for NVMe-oF with ONTAP storage

The SUSE Linux Enterprise Server 15 SPx host supports the NVMe over Fibre Channel (NVMe/FC) and NVMe over TCP (NVMe/TCP) protocols with Asymmetric Namespace Access (ANA). ANA provides multipathing functionality equivalent to asymmetric logical

unit access (ALUA) in iSCSI and FCP environments.

Learn how to configure NVMe over Fabrics (NVMe-oF) hosts for SUSE Linux Enterprise Server 15 SPx. For more support and feature information, see ONTAP support and features.

NVMe-oF with SUSE Linux Enterprise Server 15 SPx has the following known limitations:

- The `nvme disconnect-all` command disconnects both root and data filesystems and might lead to system instability. Do not issue this on systems booting from SAN over NVMe-TCP or NVMe-FC namespaces.
- NetApp sanlun host utility support isn't available for NVMe-oF. Instead, you can rely on the NetApp plug-in included in the native `nvme-cli` package for all NVMe-oF transports.
- For SUSE Linux Enterprise Server 15 SP6 and earlier, SAN booting using the NVMe-oF protocol is not supported.

Step 1: Optionally, enable SAN booting

You can configure your host to use SAN booting to simplify deployment and improve scalability. Use the Interoperability Matrix Tool to verify that your Linux OS, host bus adapter (HBA), HBA firmware, HBA boot BIOS, and ONTAP version support SAN booting.

Steps

1. Create a NVMe namespace and map it to the host.
2. Enable SAN booting in the server BIOS for the ports to which the SAN boot namespace is mapped.

   For information on how to enable the HBA BIOS, see your vendor-specific documentation.

3. Reboot the host and verify that the OS is up and running.

Step 2: Install SUSE Linux Enterprise Server and NVMe software and verify your configuration

To configure your host for NVMe-oF you need to install the host and NVMe software packages, enable multipathing, and verify your host NQN configuration.

Steps

1. Install SUSE Linux Enterprise Server 15 SPx on the server. After the installation is complete, verify that you are running the specified SUSE Linux Enterprise Server 15 SPx kernel:

   ```
   uname -r
   ```

   Example Rocky Linux kernel version:

   ```
   6.4.0-150700.53.3-default
   ```

2. Install the `nvme-cli` package:

   ```
   rpm -qa|grep nvme-cli
   ```

The following example shows an `nvme-cli` package version:

```
nvme-cli-2.11+22.gd31b1a01-150700.3.3.2.x86_64
```

3. Install the `libnvme` package:

```
rpm -qa|grep libnvme
```

The following example shows an `libnvme` package version:

```
libnvme1-1.11+4.ge68a91ae-150700.4.3.2.x86_64
```

4. On the host, check the hostnqn string at `/etc/nvme/hostnqn`:

```
cat /etc/nvme/hostnqn
```

The following example shows a `hostnqn` version:

```
nqn.2014-08.org.nvmexpress:uuid:f6517cae-3133-11e8-bbff-7ed30aef123f
```

5. On the ONTAP system, verify that the `hostnqn` string matches the `hostnqn` string for the corresponding subsystem on the ONTAP array:

```
::> vserver nvme subsystem host show -vserver vs_coexistence_LPE36002
```

Show example

```
Vserver Subsystem Priority  Host NQN
------- --------- --------
-------------------------------------------------
vs_coexistence_LPE36002
        nvme
                  regular   nqn.2014-
08.org.nvmexpress:uuid:4c4c4544-0056-5410-8048-b9c04f425633
        nvme_1
                  regular   nqn.2014-
08.org.nvmexpress:uuid:4c4c4544-0056-5410-8048-b9c04f425633
        nvme_2
                  regular   nqn.2014-
08.org.nvmexpress:uuid:4c4c4544-0056-5410-8048-b9c04f425633
        nvme_3
                  regular   nqn.2014-
08.org.nvmexpress:uuid:4c4c4544-0056-5410-8048-b9c04f425633
4 entries were displayed.
```

If the `hostnqn` strings do not match, use the `vserver modify` command to update the `hostnqn` string on your corresponding ONTAP array subsystem to match the `hostnqn` string from `/etc/nvme/hostnqn` on the host.

Step 3: Configure NVMe/FC and NVMe/TCP

Configure NVMe/FC with Broadcom/Emulex or Marvell/QLogic adapters, or configure NVMe/TCP using manual discovery and connect operations.

NVMe/FC - Broadcom/Emulex

Configure NVMe/FC for a Broadcom/Emulex FC adapter.

Steps

1. Verify that you are using the supported adapter model:
   a. Display the model names:

```
cat /sys/class/scsi_host/host*/modelname
```

   You should see the following output:

```
LPe36002-M64
LPe36002-M64
```

   b. Display the model descriptions:

```
cat /sys/class/scsi_host/host*/modeldesc
```

   You should see the following output:

```
Emulex LightPulse LPe36002-M64 2-Port 64Gb Fibre Channel Adapter
Emulex LightPulse LPe36002-M64 2-Port 64Gb Fibre Channel Adapter
```

2. Verify that you are using the recommended Broadcom `lpfc` firmware and inbox driver:
   a. Display the firmware version:

```
cat /sys/class/scsi_host/host*/fwrev
```

   The following example shows firmware versions:

```
14.4.393.25, sli-4:2:c
14.4.393.25, sli-4:2:c
```

   b. Display the inbox driver version:

```
cat /sys/module/lpfc/version
```

   The following example shows a driver version:

```
0:14.4.0.8
```

For the current list of supported adapter driver and firmware versions, see the Interoperability Matrix Tool.

3. Verify that the expected output of `lpfc_enable_fc4_type` is set to `3`:

```
cat /sys/module/lpfc/parameters/lpfc_enable_fc4_type
```

4. Verify that you can view your initiator ports:

```
cat /sys/class/fc_host/host*/port_name
```

You should see an output similar to:

```
0x10000090fae0ec88
0x10000090fae0ec89
```

5. Verify that your initiator ports are online:

```
cat /sys/class/fc_host/host*/port_state
```

You should see the following output:

```
Online
Online
```

6. Verify that the NVMe/FC initiator ports are enabled and that the target ports are visible:

```
cat /sys/class/scsi_host/host*/nvme_info
```

Show example output

```
NVME Initiator Enabled
XRI Dist lpfc0 Total 6144 IO 5894 ELS 250
NVME LPORT lpfc0 WWPN x10000090fae0ec88 WWNN x20000090fae0ec88
DID x0a1300 ONLINE
NVME RPORT       WWPN x23b1d039ea359e4a WWNN x23aed039ea359e4a
DID x0a1c01 TARGET DISCSRVC ONLINE
NVME RPORT       WWPN x22bbd039ea359e4a WWNN x22b8d039ea359e4a
DID x0a1c0b TARGET DISCSRVC ONLINE
NVME RPORT       WWPN x2362d039ea359e4a WWNN x234ed039ea359e4a
DID x0a1c10 TARGET DISCSRVC ONLINE
NVME RPORT       WWPN x23afd039ea359e4a WWNN x23aed039ea359e4a
DID x0a1a02 TARGET DISCSRVC ONLINE
NVME RPORT       WWPN x22b9d039ea359e4a WWNN x22b8d039ea359e4a
DID x0a1a0b TARGET DISCSRVC ONLINE
NVME RPORT       WWPN x2360d039ea359e4a WWNN x234ed039ea359e4a
DID x0a1a11 TARGET DISCSRVC ONLINE

NVME Statistics
LS: Xmt 0000004ea0 Cmpl 0000004ea0 Abort 00000000
LS XMIT: Err 00000000  CMPL: xb 00000000 Err 00000000
Total FCP Cmpl 0000000000102c35 Issue 0000000000102c2d OutIO
fffffffffffffff8
        abort 00000175 noxri 00000000 nondlp 0000021d qdepth
00000000 wqerr 00000007 err 00000000
FCP CMPL: xb 00000175 Err 0000058b

NVME Initiator Enabled
XRI Dist lpfc1 Total 6144 IO 5894 ELS 250
NVME LPORT lpfc1 WWPN x10000090fae0ec89 WWNN x20000090fae0ec89
DID x0a1200 ONLINE
NVME RPORT       WWPN x23b2d039ea359e4a WWNN x23aed039ea359e4a
DID x0a1d01 TARGET DISCSRVC ONLINE
NVME RPORT       WWPN x22bcd039ea359e4a WWNN x22b8d039ea359e4a
DID x0a1d0b TARGET DISCSRVC ONLINE
NVME RPORT       WWPN x2363d039ea359e4a WWNN x234ed039ea359e4a
DID x0a1d10 TARGET DISCSRVC ONLINE
NVME RPORT       WWPN x23b0d039ea359e4a WWNN x23aed039ea359e4a
DID x0a1b02 TARGET DISCSRVC ONLINE
NVME RPORT       WWPN x22bad039ea359e4a WWNN x22b8d039ea359e4a
DID x0a1b0b TARGET DISCSRVC ONLINE
NVME RPORT       WWPN x2361d039ea359e4a WWNN x234ed039ea359e4a
DID x0a1b11 TARGET DISCSRVC ONLINE

NVME Statistics
```

```
LS: Xmt 0000004e31 Cmpl 0000004e31 Abort 00000000
LS XMIT: Err 00000000  CMPL: xb 00000000 Err 00000000
Total FCP Cmpl 00000000001017f2 Issue 00000000001017ef OutIO
fffffffffffffffd
        abort 0000018a noxri 00000000 nondlp 0000012e qdepth
00000000 wqerr 00000004 err 00000000
FCP CMPL: xb 0000018a Err 000005ca
```

NVMe/FC - Marvell/QLogic

Configure NVMe/FC for a Marvell/QLogic adapter.

Steps

1. Verify that you are running the supported adapter driver and firmware versions:

```
cat /sys/class/fc_host/host*/symbolic_name
```

The follow example shows driver and firmware versions:

```
QLE2742 FW:v9.14.00 DVR:v10.02.09.400-k-debug
QLE2742 FW:v9.14.00 DVR:v10.02.09.400-k-debug
```

2. Verify that `ql2xnvmeenable` is set. This enables the Marvell adapter to function as an NVMe/FC initiator:

```
cat /sys/module/qla2xxx/parameters/ql2xnvmeenable
```

The expected output is 1.

NVMe/TCP

The NVMe/TCP protocol doesn't support the auto-connect operation. Instead, you can discover the NVMe/TCP subsystems and namespaces by performing the NVMe/TCP `connect` or `connect-all` operations manually.

Steps

1. Verify that the initiator port can fetch the discovery log page data across the supported NVMe/TCP LIFs:

```
nvme discover -t tcp -w <host-traddr> -a <traddr>
```

Show example output

```
nvme discover -t tcp -w 192.168.111.80 -a 192.168.111.70

Discovery Log Number of Records 8, Generation counter 42
=====Discovery Log Entry 0======
trtype:  tcp
adrfam:  ipv4
subtype: current discovery subsystem
treq:    not specified
portid:  4
trsvcid: 8009
subnqn:  nqn.1992-
08.com.netapp:sn.f8e2af201b7211f0ac2bd039eab67a95:discovery
traddr:  192.168.211.71
eflags:  explicit discovery connections, duplicate discovery
information
sectype: none
=====Discovery Log Entry 1======
trtype:  tcp
adrfam:  ipv4
subtype: current discovery subsystem
treq:    not specified
portid:  3
trsvcid: 8009
subnqn:  nqn.1992-
08.com.netapp:sn.f8e2af201b7211f0ac2bd039eab67a95:discovery
traddr:  192.168.111.71
eflags:  explicit discovery connections, duplicate discovery
information
sectype: none
=====Discovery Log Entry 2======
trtype:  tcp
adrfam:  ipv4
subtype: current discovery subsystem
treq:    not specified
portid:  2
trsvcid: 8009
subnqn:  nqn.1992-
08.com.netapp:sn.f8e2af201b7211f0ac2bd039eab67a95:discovery
traddr:  192.168.211.70
eflags:  explicit discovery connections, duplicate discovery
information
sectype: none
=====Discovery Log Entry 3======
trtype:  tcp
```

```
adrfam:  ipv4
subtype: current discovery subsystem
treq:    not specified
portid:  1
trsvcid: 8009
subnqn:  nqn.1992-
08.com.netapp:sn.f8e2af201b7211f0ac2bd039eab67a95:discovery
traddr:  192.168.111.70
eflags:  explicit discovery connections, duplicate discovery
information
sectype: none
=====Discovery Log Entry 4======
trtype:  tcp
adrfam:  ipv4
subtype: nvme subsystem
treq:    not specified
portid:  4
trsvcid: 4420
subnqn:  nqn.1992-
08.com.netapp:sn.f8e2af201b7211f0ac2bd039eab67a95:subsystem.samp
le_tcp_sub
traddr:  192.168.211.71
eflags:  none
sectype: none
=====Discovery Log Entry 5======
trtype:  tcp
adrfam:  ipv4
subtype: nvme subsystem
treq:    not specified
portid:  3
trsvcid: 4420
subnqn:  nqn.1992-
08.com.netapp:sn.f8e2af201b7211f0ac2bd039eab67a95:subsystem.samp
le_tcp_sub
traddr:  192.168.111.71
eflags:  none
sectype: none
=====Discovery Log Entry 6======
trtype:  tcp
adrfam:  ipv4
subtype: nvme subsystem
treq:    not specified
portid:  2
trsvcid: 4420
subnqn:  nqn.1992-
08.com.netapp:sn.f8e2af201b7211f0ac2bd039eab67a95:subsystem.samp
```

```
le_tcp_sub
traddr:  192.168.211.70
eflags:  none
sectype: none
=====Discovery Log Entry 7======
trtype:  tcp
adrfam:  ipv4
subtype: nvme subsystem
treq:    not specified
portid:  1
trsvcid: 4420
subnqn:  nqn.1992-
08.com.netapp:sn.f8e2af201b7211f0ac2bd039eab67a95:subsystem.samp
le_tcp_sub
traddr:  192.168.111.70
eflags:  none
sectype: none
localhost:~ #
```

2. Verify that all other NVMe/TCP initiator-target LIF combinations can successfully fetch discovery log page data:

```
nvme discover -t tcp -w <host-traddr> -a <traddr>
```

Show example

```
nvme discover -t tcp -w 192.168.111.80 -a 192.168.111.66
nvme discover -t tcp -w 192.168.111.80 -a 192.168.111.67
nvme discover -t tcp -w 192.168.211.80 -a 192.168.211.66
nvme discover -t tcp -w 192.168.211.80 -a 192.168.211.67
```

3. Run the `nvme connect-all` command across all the supported NVMe/TCP initiator-target LIFs across the nodes:

```
nvme connect-all -t tcp -w <host-traddr> -a <traddr>
```

Show example

```
nvme    connect-all -t  tcp -w  192.168.111.80  -a
192.168.111.66
nvme    connect-all -t  tcp -w  192.168.111.80  -a
192.168.111.67
nvme    connect-all -t  tcp -w  192.168.211.80  -a
192.168.211.66
nvme    connect-all -t  tcp -w  192.168.211.80  -a
192.168.211.67
```

Beginning with SUSE Linux Enterprise Server 15 SP6, the setting for the NVMe/TCP `ctrl_loss_tmo timeout` is automatically set to "off". As a result:

- There are no limits on the number of retries (indefinite retry).
- You don't need to manually configure a specific `ctrl_loss_tmo timeout` duration when using the `nvme connect` or `nvme connect-all` commands (option -l).
- The NVMe/TCP controllers don't experience timeouts in the event of a path failure and remain connected indefinitely.

Step 4: Optionally, modify the iopolicy in the udev rules

Beginning with SUSE Linux Enterprise Server 15 SP6, the default iopolicy for NVMe-oF is set to `round-robin`. If you want to change the iopolicy to `queue-depth`, modify the udev rules file as follows:

Steps

1. Open the udev rules file in a text editor with root privileges:

   ```
   /usr/lib/udev/rules.d/71-nvmf-netapp.rules
   ```

   You should see the following output:

   ```
   vi /usr/lib/udev/rules.d/71-nvmf-netapp.rules
   ```

2. Find the line that sets iopolicy for the NetApp ONTAP Controller, as shown in the following example rule:

   ```
   ACTION=="add", SUBSYSTEM=="nvme-subsystem", ATTR{subsystype}=="nvm",
   ATTR{model}=="NetApp ONTAP Controller", ATTR{iopolicy}="round-robin"
   ```

3. Modify the rule so that `round-robin` becomes `queue-depth`:

```
ACTION=="add", SUBSYSTEM=="nvme-subsystem", ATTR{subsystype}=="nvm",
ATTR{model}=="NetApp ONTAP Controller", ATTR{iopolicy}="queue-depth"
```

4. Reload the udev rules and apply the changes:

```
udevadm control --reload
udevadm trigger --subsystem-match=nvme-subsystem
```

5. Verify the current iopolicy for your subsystem. Replace <subsystem>, for example, `nvme-subsys0`.

```
cat /sys/class/nvme-subsystem/<subsystem>/iopolicy
```

You should see the following output:

```
queue-depth.
```

The new iopolicy applies automatically to matching NetApp ONTAP Controller devices. You don't need to reboot.

Step 5: Optionally, enable 1MB I/O for NVMe/FC

ONTAP reports a Max Data Transfer Size (MDTS) of 8 in the Identify Controller data. This means the maximum I/O request size can be up to 1MB. To issue I/O requests of size 1MB for a Broadcom NVMe/FC host, you should increase the `lpfc` value of the `lpfc_sg_seg_cnt` parameter to 256 from the default value of 64.

These steps don't apply to Qlogic NVMe/FC hosts.

Steps

1. Set the `lpfc_sg_seg_cnt` parameter to 256:

```
cat /etc/modprobe.d/lpfc.conf
```

You should see an output similar to the following example:

```
options lpfc lpfc_sg_seg_cnt=256
```

2. Run the `dracut -f` command, and reboot the host.
3. Verify that the value for `lpfc_sg_seg_cnt` is 256:

```
cat /sys/module/lpfc/parameters/lpfc_sg_seg_cnt
```

Step 6: Verify NVMe boot services

The `nvmefc-boot-connections.service` and `nvmf-autoconnect.service` boot services included in the NVMe/FC `nvme-cli` package are automatically enabled when the system boots.

After booting completes, verify that the `nvmefc-boot-connections.service` and `nvmf-autoconnect.service` boot services are enabled.

Steps

1. Verify that `nvmf-autoconnect.service` is enabled:

```
systemctl status nvmf-autoconnect.service
```

Show example output

```
nvmf-autoconnect.service - Connect NVMe-oF subsystems automatically
during boot
  Loaded: loaded (/usr/lib/systemd/system/nvmf-autoconnect.service;
enabled; preset: enabled)
  Active: inactive (dead) since Fri 2025-07-04 23:56:38 IST; 4 days
ago
  Main PID: 12208 (code=exited, status=0/SUCCESS)
    CPU: 62ms

Jul 04 23:56:26 localhost systemd[1]: Starting Connect NVMe-oF
subsystems automatically during boot...
Jul 04 23:56:38 localhost systemd[1]: nvmf-autoconnect.service:
Deactivated successfully.
Jul 04 23:56:38 localhost systemd[1]: Finished Connect NVMe-oF
subsystems automatically during boot.
```

2. Verify that `nvmefc-boot-connections.service` is enabled:

```
systemctl status nvmefc-boot-connections.service
```

Show example output

```
nvmefc-boot-connections.service - Auto-connect to subsystems on FC-
NVME devices found during boot
    Loaded: loaded (/usr/lib/systemd/system/nvmefc-boot-
connections.service; enabled; preset: enabled)
    Active: inactive (dead) since Mon 2025-07-07 19:52:30 IST; 1 day
4h ago
  Main PID: 2945 (code=exited, status=0/SUCCESS)
      CPU: 14ms

Jul 07 19:52:30 HP-DL360-14-168 systemd[1]: Starting Auto-connect to
subsystems on FC-NVME devices found during boot...
Jul 07 19:52:30 HP-DL360-14-168 systemd[1]: nvmefc-boot-
connections.service: Deactivated successfully.
Jul 07 19:52:30 HP-DL360-14-168 systemd[1]: Finished Auto-connect to
subsystems on FC-NVME devices found during boot.
```

Step 7: Verify the multipathing configuration

Verify that the in-kernel NVMe multipath status, ANA status, and ONTAP namespaces are correct for the NVMe-oF configuration.

Steps

1. Verify that the in-kernel NVMe multipath is enabled:

   ```
   cat /sys/module/nvme_core/parameters/multipath
   ```

   You should see the following output:

   ```
   Y
   ```

2. Verify that the appropriate NVMe-oF settings (such as, model set to NetApp ONTAP Controller and load balancing iopolicy set to queue-depth) for the respective ONTAP namespaces correctly reflect on the host:

   a. Display the subsystems:

      ```
      cat /sys/class/nvme-subsystem/nvme-subsys*/model
      ```

      You should see the following output:

```
NetApp ONTAP Controller
NetApp ONTAP Controller
```

b. Display the policy:

```
cat /sys/class/nvme-subsystem/nvme-subsys*/iopolicy
```

You should see the following output:

```
queue-depth
queue-depth
```

3. Verify that the namespaces are created and correctly discovered on the host:

```
nvme list
```

Show example

```
Node         SN                   Model
------------------------------------------------------------
/dev/nvme4n1 81Ix2BVuekWcAAAAAAAB   NetApp ONTAP Controller


Namespace Usage    Format             FW             Rev
------------------------------------------------------------
1                 21.47 GB / 21.47 GB   4 KiB + 0 B   FFFFFFFF
```

4. Verify that the controller state of each path is live and has the correct ANA status:

NVMe/FC

```
nvme list-subsys /dev/nvme4n5
```

Show example output

```
nvme-subsys114 - NQN=nqn.1992-
08.com.netapp:sn.9e30b9760a4911f08c87d039eab67a95:subsystem.sles
_161_27
                 hostnqn=nqn.2014-
08.org.nvmexpress:uuid:f6517cae-3133-11e8-bbff-7ed30aef123f
iopolicy=round-robin\
+- nvme114 fc traddr=nn-0x234ed039ea359e4a:pn-
0x2360d039ea359e4a,host_traddr=nn-0x20000090fae0ec88:pn-
0x10000090fae0ec88 live optimized
+- nvme115 fc traddr=nn-0x234ed039ea359e4a:pn-
0x2362d039ea359e4a,host_traddr=nn-0x20000090fae0ec88:pn-
0x10000090fae0ec88 live non-optimized
+- nvme116 fc traddr=nn-0x234ed039ea359e4a:pn-
0x2361d039ea359e4a,host_traddr=nn-0x20000090fae0ec89:pn-
0x10000090fae0ec89 live optimized
+- nvme117 fc traddr=nn-0x234ed039ea359e4a:pn-
0x2363d039ea359e4a,host_traddr=nn-0x20000090fae0ec89:pn-
0x10000090fae0ec89 live non-optimized
```

NVMe/TCP

```
nvme list-subsys /dev/nvme9n1
```

Show example output

```
nvme-subsys9 - NQN=nqn.1992-
08.com.netapp:sn.f8e2af201b7211f0ac2bd039eab67a95:subsystem.with
_inband_with_json hostnqn=nqn.2014-
08.org.nvmexpress:uuid:4c4c4544-0035-5910-804b-b2c04f444d33
iopolicy=round-robin
\
+- nvme10 tcp
traddr=192.168.111.71,trsvcid=4420,src_addr=192.168.111.80 live
non-optimized
 +- nvme11 tcp
traddr=192.168.211.70,trsvcid=4420,src_addr=192.168.211.80 live
optimized
 +- nvme12 tcp
traddr=192.168.111.70,trsvcid=4420,src_addr=192.168.111.80 live
optimized
 +- nvme9 tcp
traddr=192.168.211.71,trsvcid=4420,src_addr=192.168.211.80 live
non-optimized
```

5. Verify that the NetApp plug-in displays the correct values for each ONTAP namespace device:

Column

```
nvme netapp ontapdevices -o column
```

Show example

```
Device           Vserver                   Namespace Path
NSID UUID                                   Size
---------------- ------------------------
------------------------------------------------ ----
---------------------------------- ---------
/dev/nvme0n1     vs_161
/vol/fc_nvme_vol1/fc_nvme_ns1                       1
32fd92c7-0797-428e-a577-fdb3f14d0dc3   5.37GB
```

JSON

```
nvme netapp ontapdevices -o json
```

Show example

```
{
      "Device":"/dev/nvme98n2",
      "Vserver":"vs_161",
      "Namespace_Path":"/vol/fc_nvme_vol71/fc_nvme_ns71",
      "NSID":2,
      "UUID":"39d634c4-a75e-4fbd-ab00-3f9355a26e43",
      "LBA_Size":4096,
      "Namespace_Size":5368709120,
      "UsedBytes":430649344,
    }
  ]
}
```

Step 8: Create a persistent discovery controller

You can create a persistent discovery controller (PDC) for a SUSE Linux Enterprise Server 15 SPx host. A PDC is required to automatically detect an NVMe subsystem add or remove operation and changes to the discovery log page data.

Steps

1. Verify that the discovery log page data is available and can be retrieved through the initiator port and target LIF combination:

```
nvme discover -t <trtype> -w <host-traddr> -a <traddr>
```

Show example output

```
Discovery Log Number of Records 8, Generation counter 18
=====Discovery Log Entry 0======
trtype:  tcp
adrfam:  ipv4
subtype: current discovery subsystem
treq:    not specified
portid:  4
trsvcid: 8009
subnqn:  nqn.1992-
08.com.netapp:sn.4f7af2bd221811f0afadd039eab0dadd:discovery
traddr:  192.168.111.66
eflags:  explicit discovery connections, duplicate discovery
information
sectype: none
=====Discovery Log Entry 1======
trtype:  tcp
adrfam:  ipv4
subtype: current discovery subsystem
treq:    not specified
portid:  2
trsvcid: 8009
subnqn:  nqn.1992-
08.com.netapp:sn.4f7af2bd221811f0afadd039eab0dadd:discovery
traddr:  192.168.211.66
eflags:  explicit discovery connections, duplicate discovery
information
sectype: none
=====Discovery Log Entry 2======
trtype:  tcp
adrfam:  ipv4
subtype: current discovery subsystem
treq:    not specified
portid:  3
trsvcid: 8009
subnqn:  nqn.1992-
08.com.netapp:sn.4f7af2bd221811f0afadd039eab0dadd:discovery
traddr:  192.168.111.67
eflags:  explicit discovery connections, duplicate discovery
information
sectype: none
=====Discovery Log Entry 3======
trtype:  tcp
adrfam:  ipv4
subtype: current discovery subsystem
```

```
treq:    not specified
portid:  1
trsvcid: 8009
subnqn:  nqn.1992-
08.com.netapp:sn.4f7af2bd221811f0afadd039eab0dadd:discovery
traddr:  192.168.211.67
eflags:  explicit discovery connections, duplicate discovery
information
sectype: none
=====Discovery Log Entry 4======
trtype:  tcp
adrfam:  ipv4
subtype: nvme subsystem
treq:    not specified
portid:  4
trsvcid: 4420
subnqn:  nqn.1992-
08.com.netapp:sn.4f7af2bd221811f0afadd039eab0dadd:subsystem.pdc
traddr:  192.168.111.66
eflags:  none
sectype: none
=====Discovery Log Entry 5======
trtype:  tcp
adrfam:  ipv4
subtype: nvme subsystem
treq:    not specified
portid:  2
trsvcid: 4420
subnqn:  nqn.1992-
08.com.netapp:sn.4f7af2bd221811f0afadd039eab0dadd:subsystem.pdc
traddr:  192.168.211.66
eflags:  none
sectype: none
=====Discovery Log Entry 6======
trtype:  tcp
adrfam:  ipv4
subtype: nvme subsystem
treq:    not specified
portid:  3
trsvcid: 4420
subnqn:  nqn.1992-
08.com.netapp:sn.4f7af2bd221811f0afadd039eab0dadd:subsystem.pdc
traddr:  192.168.111.67
eflags:  none
sectype: none
=====Discovery Log Entry 7======
```

```
trtype:  tcp
adrfam:  ipv4
subtype: nvme subsystem
treq:    not specified
portid:  1
trsvcid: 4420
subnqn:  nqn.1992-
08.com.netapp:sn.4f7af2bd221811f0afadd039eab0dadd:subsystem.pdc
traddr:  192.168.211.67
eflags:  none
sectype: none
```

2. Create a PDC for the discovery subsystem:

```
nvme discover -t <trtype> -w <host-traddr> -a <traddr> -p
```

You should see the following output:

```
nvme discover -t tcp -w 192.168.111.80 -a 192.168.111.66 -p
```

3. From the ONTAP controller, verify that the PDC has been created:

```
vserver nvme show-discovery-controller -instance -vserver <vserver_name>
```

Show example output

```
vserver nvme show-discovery-controller -instance -vserver vs_pdc

           Vserver Name: vs_pdc
               Controller ID: 0101h
     Discovery Subsystem NQN: nqn.1992-
08.com.netapp:sn.4f7af2bd221811f0afadd039eab0dadd:discovery
           Logical Interface: lif2
                        Node: A400-12-181
                    Host NQN: nqn.2014-
08.org.nvmexpress:uuid:9796c1ec-0d34-11eb-b6b2-3a68dd3bab57
          Transport Protocol: nvme-tcp
 Initiator Transport Address: 192.168.111.80
Transport Service Identifier: 8009
             Host Identifier: 9796c1ec0d3411ebb6b23a68dd3bab57
            Admin Queue Depth: 32
       Header Digest Enabled: false
         Data Digest Enabled: false
   Keep-Alive Timeout (msec): 30000
```

Step 9: Set up secure in-band authentication

Secure in-band authentication is supported over NVMe/TCP between a SUSE Linux Enterprise Server 15 SPx host and an ONTAP controller.

Each host or controller must be associated with a `DH-HMAC-CHAP` key to set up secure authentication. A DH-HMAC-CHAP key is a combination of the NQN of the NVMe host or controller and an authentication secret configured by the administrator. To authenticate its peer, an NVMe host or controller must recognize the key associated with the peer.

Steps

Set up secure in-band authentication using the CLI or a config JSON file. If you need to specify different dhchap keys for different subsystems, you must use a config JSON file.

CLI

Set up secure in-band authentication using the CLI.

1. Obtain the host NQN:

```
cat /etc/nvme/hostnqn
```

2. Generate the dhchap key for the host.

   The following output describes the `gen-dhchap-key` command paramters:

```
nvme gen-dhchap-key -s optional_secret -l key_length {32|48|64} -m
HMAC_function {0|1|2|3} -n host_nqn
•   -s secret key in hexadecimal characters to be used to initialize
the host key
•   -l length of the resulting key in bytes
•   -m HMAC function to use for key transformation
0 = none, 1- SHA-256, 2 = SHA-384, 3=SHA-512
•   -n host NQN to use for key transformation
```

   In the following example, a random dhchap key with HMAC set to 3 (SHA-512) is generated.

```
nvme gen-dhchap-key -m 3 -n nqn.2014-
08.org.nvmexpress:uuid:e6dade64-216d-11ec-b7bb-7ed30a5482c3
DHHC-
1:03:1CFivw9ccz58gAcOUJrM7Vs98hd2ZHSr+iw+Amg6xZPl5D2Yk+HDTZiUAg1iGgx
TYqnxukqvYedA55Bw3wtz6sJNpR4=:
```

3. On the ONTAP controller, add the host and specify both dhchap keys:

```
vserver nvme subsystem host add -vserver <svm_name> -subsystem
<subsystem> -host-nqn <host_nqn> -dhchap-host-secret
<authentication_host_secret> -dhchap-controller-secret
<authentication_controller_secret> -dhchap-hash-function {sha-
256|sha-512} -dhchap-group {none|2048-bit|3072-bit|4096-bit|6144-
bit|8192-bit}
```

4. A host supports two types of authentication methods, unidirectional and bidirectional. On the host, connect to the ONTAP controller and specify dhchap keys based on the chosen authentication method:

```
nvme connect -t tcp -w <host-traddr> -a <tr-addr> -n <host_nqn> -S
<authentication_host_secret> -C <authentication_controller_secret>
```

5. Validate the `nvme connect authentication` command by verifying the host and controller dhchap keys:
   a. Verify the host dhchap keys:

```
cat /sys/class/nvme-subsystem/<nvme-subsysX>/nvme*/dhchap_secret
```

Show example output for a unidirectional configuration

```
cat /sys/class/nvme-subsystem/nvme-subsys1/nvme*/dhchap_secret
DHHC-1:01:iM63E6cX7G5SOKKOju8gmzM53qywsy+C/YwtzxhIt9ZRz+ky:
DHHC-1:01:iM63E6cX7G5SOKKOju8gmzM53qywsy+C/YwtzxhIt9ZRz+ky:
DHHC-1:01:iM63E6cX7G5SOKKOju8gmzM53qywsy+C/YwtzxhIt9ZRz+ky:
DHHC-1:01:iM63E6cX7G5SOKKOju8gmzM53qywsy+C/YwtzxhIt9ZRz+ky:
```

   b. Verify the controller dhchap keys:

```
cat /sys/class/nvme-subsystem/<nvme-
subsysX>/nvme*/dhchap_ctrl_secret
```

Show example output for a bidirectional configuration

```
cat /sys/class/nvme-subsystem/nvme-
subsys6/nvme*/dhchap_ctrl_secret
DHHC-
1:03:1CFivw9ccz58gAcOUJrM7Vs98hd2ZHSr+iw+Amg6xZPl5D2Yk+HDTZiUA
g1iGgxTYqnxukqvYedA55Bw3wtz6sJNpR4=:
DHHC-
1:03:1CFivw9ccz58gAcOUJrM7Vs98hd2ZHSr+iw+Amg6xZPl5D2Yk+HDTZiUA
g1iGgxTYqnxukqvYedA55Bw3wtz6sJNpR4=:
DHHC-
1:03:1CFivw9ccz58gAcOUJrM7Vs98hd2ZHSr+iw+Amg6xZPl5D2Yk+HDTZiUA
g1iGgxTYqnxukqvYedA55Bw3wtz6sJNpR4=:
DHHC-
1:03:1CFivw9ccz58gAcOUJrM7Vs98hd2ZHSr+iw+Amg6xZPl5D2Yk+HDTZiUA
g1iGgxTYqnxukqvYedA55Bw3wtz6sJNpR4=:
```

JSON

When multiple NVMe subsystems are available on the ONTAP controller configuration, you can use the `/etc/nvme/config.json` file with the `nvme connect-all` command.

Use the `-o` option to generate the JSON file. Refer to the NVMe connect-all man pages for more syntax options.

1. Configure the JSON file:

Show example output

```
cat /etc/nvme/config.json
[
 {
    "hostnqn":"nqn.2014-08.org.nvmexpress:uuid:4c4c4544-0035-
5910-804b-b2c04f444d33",
    "hostid":"4c4c4544-0035-5910-804b-b2c04f444d33",
    "dhchap_key":"DHHC-
1:01:i4i789R11sMuHLCY27RVI8XloC\/GzjRwyhxip5hmIELsHrBq:",
    "subsystems":[
      {
        "nqn":"nqn.1992-
08.com.netapp:sn.f8e2af201b7211f0ac2bd039eab67a95:subsystem.samp
le_tcp_sub",
        "ports":[
          {
            "transport":"tcp",
            "traddr":"192.168.111.70",
            "host_traddr":"192.168.111.80",
            "trsvcid":"4420"
            "dhchap_ctrl_key":"DHHC-
1:03:jqgYcJSKp73+XqAf2X6twr9ngBpr2n0MGWbmZIZq4PieKZCoilKGef8lAvh
YS0PNK7T+04YD5CRPjh+m3qjJU++yR8s=:"
          },
               {
                    "transport":"tcp",
                    "traddr":"192.168.111.71",
                    "host_traddr":"192.168.111.80",
                    "trsvcid":"4420",
                    "dhchap_ctrl_key":"DHHC-
1:03:jqgYcJSKp73+XqAf2X6twr9ngBpr2n0MGWbmZIZq4PieKZCoilKGef8lAvh
YS0PNK7T+04YD5CRPjh+m3qjJU++yR8s=:"
               },
               {
                    "transport":"tcp",
                    "traddr":"192.168.211.70",
                    "host_traddr":"192.168.211.80",
                    "trsvcid":"4420",
                    "dhchap_ctrl_key":"DHHC-
1:03:jqgYcJSKp73+XqAf2X6twr9ngBpr2n0MGWbmZIZq4PieKZCoilKGef8lAvh
YS0PNK7T+04YD5CRPjh+m3qjJU++yR8s=:"
               },
               {
                    "transport":"tcp",
                    "traddr":"192.168.211.71",
```

```
                    "host_traddr":"192.168.211.80",
                    "trsvcid":"4420",
                    "dhchap_ctrl_key":"DHHC-
1:03:jqgYcJSKp73+XqAf2X6twr9ngBpr2n0MGWbmZIZq4PieKZCoilKGef8lAvh
YS0PNK7T+04YD5CRPjh+m3qjJU++yR8s=:"
                }
            ]
        }
    ]
 }
]
```

In the following example, `dhchap_key` corresponds to `dhchap_secret` and `dhchap_ctrl_key` corresponds to `dhchap_ctrl_secret`.

2. Connect to the ONTAP controller using the config JSON file:

```
nvme connect-all -J /etc/nvme/config.json
```

Show example output

```
traddr=192.168.211.70 is already connected
traddr=192.168.111.71 is already connected
traddr=192.168.211.71 is already connected
traddr=192.168.111.70 is already connected
traddr=192.168.211.70 is already connected
traddr=192.168.111.70 is already connected
traddr=192.168.211.71 is already connected
traddr=192.168.111.71 is already connected
traddr=192.168.211.70 is already connected
traddr=192.168.111.71 is already connected
traddr=192.168.211.71 is already connected
traddr=192.168.111.70 is already connected
```

3. Verify that the dhchap secrets have been enabled for the respective controllers for each subsystem:
   a. Verify the host dhchap keys:

```
cat /sys/class/nvme-subsystem/nvme-subsys0/nvme0/dhchap_secret
```

   The following example shows a dhchap key:

```
DHHC-1:01:i4i789R11sMuHLCY27RVI8XloC/GzjRwyhxip5hmIELsHrBq:
```

b. Verify the controller dhchap keys:

```
cat /sys/class/nvme-subsystem/nvme-
subsys0/nvme0/dhchap_ctrl_secret
```

You should see an output similar to the following example:

```
DHHC-
1:03:jqgYcJSKp73+XqAf2X6twr9ngBpr2n0MGWbmZIZq4PieKZCoilKGef8lAvhY
S0PNK7T+04YD5CRPjh+m3qjJU++yR8s=:
```

Step 10: Configure Transport Layer Security

Transport Layer Security (TLS) provides secure end-to-end encryption for NVMe connections between NVMe-oF hosts and an ONTAP array. You can configure TLS 1.3 using the CLI and a configured pre-shared key (PSK).

Perform the following steps on the SUSE Linux Enterprise Server host, except where it specifies that you perform a step on the ONTAP controller.

Steps

1. Check that you have the following `ktls-utils`, `openssl`, and `libopenssl` packages installed on the host:

   a. Verify the `ktls-utils`:

   ```
   rpm -qa | grep ktls
   ```

   You should see the following output displayed:

   ```
   ktls-utils-0.10+33.g311d943-150700.1.5.x86_64
   ```

   b. Verify the SSL packages:

   ```
   rpm -qa | grep ssl
   ```

Show example output

```
libopenssl3-3.2.3-150700.3.20.x86_64
openssl-3-3.2.3-150700.3.20.x86_64
libopenssl1_1-1.1.1w-150700.9.37.x86_64
```

2. Verify that you have the correct setup for `/etc/tlshd.conf`:

```
cat /etc/tlshd.conf
```

Show example output

```
[debug]
loglevel=0
tls=0
nl=0
[authenticate]
keyrings=.nvme
[authenticate.client]
#x509.truststore= <pathname>
#x509.certificate= <pathname>
#x509.private_key= <pathname>
[authenticate.server]
#x509.truststore= <pathname>
#x509.certificate= <pathname>
#x509.private_key= <pathname>
```

3. Enable `tlshd` to start at system boot:

```
systemctl enable tlshd
```

4. Verify that the `tlshd` daemon is running:

```
systemctl status tlshd
```

Show example output

```
tlshd.service - Handshake service for kernel TLS consumers
   Loaded: loaded (/usr/lib/systemd/system/tlshd.service; enabled;
preset: disabled)
   Active: active (running) since Wed 2024-08-21 15:46:53 IST; 4h
57min ago
     Docs: man:tlshd(8)
Main PID: 961 (tlshd)
   Tasks: 1
     CPU: 46ms
   CGroup: /system.slice/tlshd.service
       └─961 /usr/sbin/tlshd
Aug 21 15:46:54 RX2530-M4-17-153 tlshd[961]: Built from ktls-utils
0.11-dev on Mar 21 2024 12:00:00
```

5. Generate the TLS PSK by using the `nvme gen-tls-key`:
   a. Verify the host:

      ```
      cat /etc/nvme/hostnqn
      ```

      You should see the following output:

      ```
      nqn.2014-08.org.nvmexpress:uuid:4c4c4544-0035-5910-804b-b2c04f444d33
      ```

   b. Verify the key:

      ```
      nvme gen-tls-key --hmac=1 --identity=1 --subsysnqn= nqn.1992-
      08.com.netapp:sn.a2d41235b78211efb57dd039eab67a95:subsystem.nvme1
      ```

      You should see the following output:

      ```
      NVMeTLSkey-1:01:C50EsaGtuOp8n5fGE9EuWjbBCtshmfoHx4XTqTJUmydf0gIj:
      ```

6. On the ONTAP controller, add the TLS PSK to the ONTAP subsystem:

Show example output

```
nvme subsystem host add -vserver vs_iscsi_tcp -subsystem nvme1 -host
-nqn nqn.2014-08.org.nvmexpress:uuid:4c4c4544-0035-5910-804b-
b2c04f444d33 -tls-configured-psk NVMeTLSkey-
1:01:C50EsaGtuOp8n5fGE9EuWjbBCtshmfoHx4XTqTJUmydf0gIj:
```

7. Insert the TLS PSK into the host kernel keyring:

```
nvme check-tls-key --identity=1 --subsysnqn=nqn.1992
-08.com.netapp:sn.a2d41235b78211efb57dd039eab67a95:subsystem.nvme1
--keydata=NVMeTLSkey
-1:01:C50EsaGtuOp8n5fGE9EuWjbBCtshmfoHx4XTqTJUmydf0gIj: --insert
```

You should see the following TLS key:

```
Inserted TLS key 22152a7e
```

> The PSK shows as `NVMe1R01` because it uses `identity v1` from the TLS handshake algorithm. Identity v1 is the only version that ONTAP supports.

8. Verify that the TLS PSK is inserted correctly:

```
cat /proc/keys | grep NVMe
```

Show example output

```
069f56bb I--Q---     5 perm 3b010000     0     0 psk       NVMe1R01
nqn.2014-08.org.nvmexpress:uuid:4c4c4544-0035-5910-804b-b2c04f444d33
nqn.1992-
08.com.netapp:sn.a2d41235b78211efb57dd039eab67a95:subsystem.nvme1
oYVLelmiOwnvDjXKBmrnIgGVpFIBDJtc4hmQXE/36Sw=: 32
```

9. Connect to the ONTAP subsystem using the inserted TLS PSK:
   a. Verify the TLS PSK:

```
nvme connect -t tcp -w 192.168.111.80 -a 192.168.111.66  -n nqn.1992-
08.com.netapp:sn.a2d41235b78211efb57dd039eab67a95:subsystem.nvme1
--tls_key=0x069f56bb –tls
```

You should see the following output:

```
connecting to device: nvme0
```

b. Verify the list-subsys:

```
nvme list-subsys
```

Show example output

```
nvme-subsys0 - NQN=nqn.1992-
08.com.netapp:sn.a2d41235b78211efb57dd039eab67a95:subsystem.nvme1
               hostnqn=nqn.2014-08.org.nvmexpress:uuid:4c4c4544-
0035-5910-804b-b2c04f444d33
\
 +- nvme0 tcp
traddr=192.168.111.66,trsvcid=4420,host_traddr=192.168.111.80,src
_addr=192.168.111.80 live
```

10. Add the target, and verify the TLS connection to the specified ONTAP subsystem:

```
nvme subsystem controller show -vserver sles15_tls -subsystem sles15
-instance
```

Show example output

```
(vserver nvme subsystem controller show)
                          Vserver Name: vs_iscsi_tcp
                          Subsystem: nvme1
                      Controller ID: 0040h
                  Logical Interface: tcpnvme_lif1_1
                               Node: A400-12-181
                           Host NQN: nqn.2014-
08.org.nvmexpress:uuid:4c4c4544-0035-5910-804b-b2c04f444d33
                 Transport Protocol: nvme-tcp
        Initiator Transport Address: 192.168.111.80
                    Host Identifier:
4c4c454400355910804bb2c04f444d33
               Number of I/O Queues: 2
                   I/O Queue Depths: 128, 128
                  Admin Queue Depth: 32
              Max I/O Size in Bytes: 1048576
          Keep-Alive Timeout (msec): 5000
                     Subsystem UUID: 8bbfb403-1602-11f0-ac2b-
d039eab67a95
              Header Digest Enabled: false
                Data Digest Enabled: false
       Authentication Hash Function: sha-256
Authentication Diffie-Hellman Group: 3072-bit
                Authentication Mode: unidirectional
       Transport Service Identifier: 4420
                       TLS Key Type: configured
                   TLS PSK Identity: NVMe1R01 nqn.2014-
08.org.nvmexpress:uuid:4c4c4544-0035-5910-804b-b2c04f444d33
nqn.1992-
08.com.netapp:sn.a2d41235b78211efb57dd039eab67a95:subsystem.nvme1
oYVLelmiOwnvDjXKBmrnIgGVpFIBDJtc4hmQXE/36Sw=
                         TLS Cipher: TLS-AES-128-GCM-SHA256
```

Step 11: Review the known issues

There are no known issues.

Ubuntu

NVMe-oF host configuration for Ubuntu 24.04 with ONTAP

NVMe over Fabrics (NVMe-oF), including NVMe over Fibre Channel (NVMe/FC) and

other transports, is supported with Ubuntu 24.04 and Asymmetric Namespace Access (ANA). In NVMe-oF environments, ANA is the equivalent of ALUA multipathing in iSCSI and FC environments and is implemented with in-kernel NVMe multipath.

The following support is available for the NVMe-oF host configuration for Ubuntu 24.04 with ONTAP:

- The NetApp plug-in in the native nvme-cli package displays ONTAP details for NVMe/FC namespaces.
- Use of NVMe and SCSI co-existent traffic on the same host on a given host bus adapter (HBA), without the explicit dm-multipath settings to prevent claiming NVMe namespaces.

For additional details on supported configurations, see the Interoperability Matrix Tool.

Features

Ubuntu 24.04 has in-kernel NVMe multipath enabled for NVMe namespaces by default. This means you don't need explicit settings.

Known limitations

SAN booting using the NVMe-oF protocol is currently not supported for Ubuntu 24.04 with ONTAP.

Validate software versions

You can use the following procedure to validate the minimum supported Ubuntu 24.04 software versions.

Steps

1. Install Ubuntu 24.04 on the server. After the installation is complete, verify that you are running the specified Ubuntu 24.04 kernel:

```
uname -r
```

```
6.8.0-31-generic
```

2. Install the `nvme-cli` package:

```
apt list | grep nvme
```

```
nvme-cli/noble-updates 2.8-1ubuntu0.1 amd64
```

3. On the Ubuntu 24.04 host, check the hostnqn string at `/etc/nvme/hostnqn`:

```
cat /etc/nvme/hostnqn
```

```
nqn.2014-08.org.nvmexpress:uuid:ace18dd8-1f5a-11ec-b0c3-3a68dd61a6ff
```

4. Verify that the `hostnqn` string from `/etc/nvme/hostnqn` matches the `hostnqn` string for the corresponding subsystem on the ONTAP array:

```
vserver nvme subsystem host show -vserver vs_106_fc_nvme
```

```
Vserver     Subsystem          Host NQN
----------- ---------------
------------------------------------------------------------
vs_106_fc_nvme ub_106 nqn.2014-08.org.nvmexpress:uuid:c04702c8-e91e-
4353-9995-ba4536214631
```

If the `hostnqn` strings don't match, use the `vserver modify` command to update the `hostnqn` string on your corresponding ONTAP array subsystem to match the `hostnqn` string from `/etc/nvme/hostnqn` on the host.

Configure NVMe/FC

You can configure NVMe/FC for Broadcom/Emulex or Marvell/Qlogic adapters.

Broadcom/Emulex

Configure NVMe/FC for a Broadcom/Emulex adapter.

1. Verify that you're using the supported adapter model:

   a. `cat /sys/class/scsi_host/host*/modelname`

   ```
   LPe36002-M64
   LPe36002-M64
   ```

   b. `cat /sys/class/scsi_host/host*/modeldesc`

   ```
   Emulex LPe36002-M64 2-Port 64Gb Fibre Channel Adapter
   Emulex LPe36002-M64 2-Port 64Gb Fibre Channel Adapter
   ```

2. Verify that you're using the recommended Broadcom `lpfc` firmware and inbox driver.

   a. `cat /sys/class/scsi_host/host*/fwrev`

   ```
   14.4.317.10, sli-4:6:d
   14.4.317.10, sli-4:6:d
   ```

   b. `cat /sys/module/lpfc/version`

   ```
   0:14.2.0.17
   ```

   For the current list of supported adapter driver and firmware versions, see the Interoperability Matrix Tool.

3. Verify that `lpfc_enable_fc4_type` is set to `3`:

   ```
   cat /sys/module/lpfc/parameters/lpfc_enable_fc4_type
   ```

   The expected output is 3.

4. Verify that the initiator ports are up and running and that you can see the target LIFs:

   a. `cat /sys/class/fc_host/host*/port_name`

   ```
   0x100000109bf0447b
   0x100000109bf0447c
   ```

b. `cat /sys/class/fc_host/host*/port_state`

```
Online
Online
```

c. `cat /sys/class/scsi_host/host*/nvme_info`

Show example output

```
NVME Initiator Enabled
XRI Dist lpfc0 Total 6144 IO 5894 ELS 250
NVME LPORT lpfc0 WWPN x100000109bf0447b WWNN x200000109bf0447b
DID x022600 ONLINE
NVME RPORT       WWPN x200fd039eaa8138b WWNN x200ad039eaa8138b
DID x021006 TARGET DISCSRVC ONLINE

NVME Statistics
LS: Xmt 0000000187 Cmpl 0000000187 Abort 00000000
LS XMIT: Err 00000000  CMPL: xb 00000000 Err 00000000
Total FCP Cmpl 0000000014096514 Issue 000000001407fcd6 OutIO
fffffffffffe97c2
        abort 00000048 noxri 00000000 nondlp 0000001c qdepth
00000000 wqerr 00000000 err 00000000
FCP CMPL: xb 00000048 Err 00000077

NVME Initiator Enabled
XRI Dist lpfc1 Total 6144 IO 5894 ELS 250
NVME LPORT lpfc1 WWPN x100000109bf0447c WWNN x200000109bf0447c
DID x022300 ONLINE
NVME RPORT       WWPN x2010d039eaa8138b WWNN x200ad039eaa8138b
DID x021106 TARGET DISCSRVC ONLINE

NVME Statistics
LS: Xmt 0000000187 Cmpl 0000000187 Abort 00000000
LS XMIT: Err 00000000  CMPL: xb 00000000 Err 00000000
Total FCP Cmpl 00000000140970ed Issue 00000000140813da OutIO
fffffffffffea2ed
        abort 00000047 noxri 00000000 nondlp 0000002b qdepth
00000000 wqerr 00000000 err 00000000
FCP CMPL: xb 00000047 Err 00000075
```

Marvell/QLogic

The native inbox qla2xxx driver included in the Ubuntu 24.04 GA kernel has the latest upstream fixes.

These fixes are essential for ONTAP support.

Configure NVMe/FC for a Marvell/QLogic adapter.

Steps

1. Verify that you are running the supported adapter driver and firmware versions:

```
cat /sys/class/fc_host/host*/symbolic_name
```

```
QLE2872 FW: v9.15.00 DVR: v10.02.09.100-k
QLE2872 FW: v9.15.00 DVR: v10.02.09.100-k
```

2. Verify that `ql2xnvmeenable` is set. This enables the Marvell adapter to function as an NVMe/FC initiator:

```
cat /sys/module/qla2xxx/parameters/ql2xnvmeenable
```

The expected ouptut is 1.

Enable 1MB I/O (Optional)

ONTAP reports a Max Data Transfer Size (MDTS) of 8 in the Identify Controller data. This means the maximum I/O request size can be up to 1MB. To issue I/O requests of size 1MB for a Broadcom NVMe/FC host, you should increase the `lpfc` value of the `lpfc_sg_seg_cnt` parameter to 256 from the default value of 64.

These steps don't apply to Qlogic NVMe/FC hosts.

Steps

1. Set the `lpfc_sg_seg_cnt` parameter to 256:

```
cat /etc/modprobe.d/lpfc.conf
```

You should see an output similar to the following example:

```
options lpfc lpfc_sg_seg_cnt=256
```

2. Run the `dracut -f` command, and reboot the host.
3. Verify that the value for `lpfc_sg_seg_cnt` is 256:

```
cat /sys/module/lpfc/parameters/lpfc_sg_seg_cnt
```

Configure NVMe/TCP

NVMe/TCP doesn’t support the auto-connect functionality. Instead, you can manually discover the NVMe/TCP subsystems and namespaces by using the `connect` or `connect-all` commands.

Steps

1. Verify that the initiator port can fetch the discovery log page data across the supported NVMe/TCP LIFs:

```
nvme discover -t tcp -w <host-traddr> -a <traddr>
```

Show example

```
# nvme discover -t tcp -w 192.168.167.150 -a 192.168.167.155
Discovery Log Number of Records 8, Generation counter 10
=====Discovery Log Entry 0======
trtype:  tcp
adrfam:  ipv4
subtype: current discovery subsystem
treq:    not specified
portid:  4
trsvcid: 8009
subnqn:  nqn.1992-
08.com.netapp:sn.9b7d42b764ff11efb8fed039eabac370:discovery
traddr:  192.168.167.156
eflags:  explicit discovery connections, duplicate discovery
information
sectype: none
=====Discovery Log Entry 1======
trtype:  tcp
adrfam:  ipv4
subtype: current discovery subsystem
treq:    not specified
portid:  2
trsvcid: 8009
subnqn:  nqn.1992-
08.com.netapp:sn.9b7d42b764ff11efb8fed039eabac370:discovery
traddr:  192.168.166.156
eflags:  explicit discovery connections, duplicate discovery
information
sectype: none
=====Discovery Log Entry 2======
trtype:  tcp
adrfam:  ipv4
subtype: current discovery subsystem
treq:    not specified
portid:  3
trsvcid: 8009
subnqn:  nqn.1992-
08.com.netapp:sn.9b7d42b764ff11efb8fed039eabac370:discovery
traddr:  192.168.167.155
eflags:  explicit discovery connections, duplicate discovery
information
sectype: none
=====Discovery Log Entry 3======
trtype:  tcp
adrfam:  ipv4
```

```
subtype: current discovery subsystem
treq:    not specified
portid:  1
trsvcid: 8009
subnqn:  nqn.1992-
08.com.netapp:sn.9b7d42b764ff11efb8fed039eabac370:discovery
traddr:  192.168.166.155
eflags:  explicit discovery connections, duplicate discovery
information
sectype: none
=====Discovery Log Entry 4======
trtype:  tcp
adrfam:  ipv4
subtype: nvme subsystem
treq:    not specified
portid:  4
trsvcid: 4420
subnqn:  nqn.1992-
08.com.netapp:sn.9b7d42b764ff11efb8fed039eabac370:subsystem.ubuntu_2
4.04_tcp_211
traddr:  192.168.167.156
eflags:  none
sectype: none
=====Discovery Log Entry 5======
trtype:  tcp
adrfam:  ipv4
subtype: nvme subsystem
treq:    not specified
portid:  2
trsvcid: 4420
subnqn:  nqn.1992-
08.com.netapp:sn.9b7d42b764ff11efb8fed039eabac370:subsystem.ubuntu_2
4.04_tcp_211
traddr:  192.168.166.156
eflags:  none
sectype: none
=====Discovery Log Entry 6======
trtype:  tcp
adrfam:  ipv4
subtype: nvme subsystem
treq:    not specified
portid:  3
trsvcid: 4420
subnqn:  nqn.1992-
08.com.netapp:sn.9b7d42b764ff11efb8fed039eabac370:subsystem.ubuntu_2
4.04_tcp_211
```

```
traddr:  192.168.167.155
eflags:  none
sectype: none
=====Discovery Log Entry 7======
trtype:  tcp
adrfam:  ipv4
subtype: nvme subsystem
treq:    not specified
portid:  1
trsvcid: 4420
subnqn:  nqn.1992-
08.com.netapp:sn.9b7d42b764ff11efb8fed039eabac370:subsystem.ubuntu_2
4.04_tcp_211
traddr:  192.168.166.155
eflags:  none
sectype: none
```

2. Verify that the other NVMe/TCP initiator-target LIF combinations can fetch discovery log page data:

```
nvme discover -t tcp -w <host-traddr> -a <traddr>
```

Show example output

```
#nvme discover -t tcp -w 192.168.167.150 -a 192.168.167.155
#nvme discover -t tcp -w 192.168.167.150 -a 192.168.167.156
#nvme discover -t tcp -w 192.168.166.150 -a 192.168.166.155
#nvme discover -t tcp -w 192.168.166.150 -a 192.168.166.156
```

3. Run the `nvme connect-all` command across all the supported NVMe/TCP initiator-target LIFs across the nodes:

```
nvme connect-all -t tcp -w <host-traddr> -a <traddr>
```

Show example output

```
#nvme connect-all -t tcp -w 192.168.167.150 -a 192.168.167.155
#nvme connect-all -t tcp -w 192.168.167.150 -a 192.168.167.156
#nvme connect-all -t tcp -w 192.168.166.150 -a 192.168.166.155
#nvme connect-all -t tcp -w 192.168.166.150 -a 192.168.166.156
```

Beginning with Ubuntu 24.04, the ctrl_loss_tmo timeout default setting for NVMe/TCP is turned off. This means there is no limit on the number of retries (indefinite retry) and you don't need to manually configure a specific ctrl_loss_tmo timeout duration when using the `nvme connect` or `nvme connect-all` commands (option -l). With this default behavior, the NVMe/TCP controllers don't experience timeouts in the event of a path failure and remain connected indefinitely.

Validate NVMe-oF

You can use the following procedure to validate NVME-oF.

Steps

1. Verify that the in-kernel NVMe multipath is enabled:

   ```
   cat /sys/module/nvme_core/parameters/multipath
   ```

   The expected output is "Y".

2. Verify that the appropriate NVMe-oF settings (such as, model set to "NetApp ONTAP Controller" and load balancing iopolicy set to "round-robin") for the respective ONTAP namespaces correctly display on the host:

   a. `cat /sys/class/nvme-subsystem/nvme-subsys*/model`

   ```
   NetApp ONTAP Controller
   NetApp ONTAP Controller
   ```

   b. `cat /sys/class/nvme-subsystem/nvme-subsys*/iopolicy`

   ```
   round-robin
   round-robin
   ```

3. Verify that the namespaces are created and correctly discovered on the host:

   ```
   nvme list
   ```

Show example output

```
Node         SN                            Model
-------------------------------------------------------
/dev/nvme0n1 81CZ5BQuUNfGAAAAAAAB   NetApp ONTAP Controller


Namespace Usage   Format                FW            Rev
-------------------------------------------------------------
1                 21.47 GB / 21.47 GB   4 KiB + 0 B   FFFFFFFF
```

4. Verify that the controller state of each path is live and has the correct ANA status:

NVMe/FC

```
nvme list-subsys /dev/nvme0n1
```

Show example output

```
nvme-subsys4 - NQN=nqn.1992-
08.com.netapp:sn.8763d311b2ac11ed950ed039ea951c46:subsystem.
ubuntu_24.04 \
+- nvme1 fc traddr=nn-0x20a6d039ea954d17:pn-
0x20a7d039ea954d17,host_traddr=nn-0x200000109b1b95ef:pn-
0x100000109b1b95ef live optimized
+- nvme2 fc traddr=nn-0x20a6d039ea954d17:pn-
0x20a8d039ea954d17,host_traddr=nn-0x200000109b1b95f0:pn-
0x100000109b1b95f0 live optimized
+- nvme3 fc traddr=nn-0x20a6d039ea954d17:pn-
0x20aad039ea954d17,host_traddr=nn-0x200000109b1b95f0:pn-
0x100000109b1b95f0 live non-optimized
+- nvme5 fc traddr=nn-0x20a6d039ea954d17:pn-
0x20a9d039ea954d17,host_traddr=nn-0x200000109b1b95ef:pn-
0x100000109b1b95ef live non-optimized
```

NVMe/TCP

```
nvme list-subsys /dev/nvme1n1
```

Show example output

```
nvme-subsys0 - NQN=nqn.1992-
08.com.netapp:sn.9b7d42b764ff11efb8fed039eabac370:subsystem.ubun
tu_24.04_tcp_211
               hostnqn=nqn.2014-08.org.nvmexpress:uuid:4c4c4544-
0050-3410-8035-c3c04f4a5933
               iopolicy=round-robin
 +- nvme0 tcp
traddr=192.168.166.155,trsvcid=4420,host_traddr=192.168.166.150,
src_addr=192.168.166.150 live optimized
 +- nvme1 tcp
traddr=192.168.167.155,trsvcid=4420,host_traddr=192.168.167.150,
src_addr=192.168.167.150 live optimized
 +- nvme2 tcp
traddr=192.168.166.156,trsvcid=4420,host_traddr=192.168.166.150,
src_addr=192.168.166.150 live non-optimized
 +- nvme3 tcp
traddr=192.168.167.156,trsvcid=4420,host_traddr=192.168.167.150,
src_addr=192.168.167.150 live non-optimized
```

5. Verify that the NetApp plug-in displays the correct values for each ONTAP namespace device:

Column

```
nvme netapp ontapdevices -o column
```

Show example output

```
Device        Vserver        Namespace Path          NSID UUID
Size
------------- -------------  ----------------------- ----
---------------------------------- ---------
/dev/nvme0n1   vs_211_tcp    /vol/tcpvol1/ns1        1
1cc7bc78-8d7b-4d8e-a3c4-750f9461a6e9   21.47GB
```

JSON

```
nvme netapp ontapdevices -o json
```

Show example output

```
{

"ONTAPdevices" : [
{
      "Device":"/dev/nvme0n9",
      "Vserver":"vs_211_tcp",
      "Namespace_Path":"/vol/tcpvol9/ns9",
      "NSID":9,
      "UUID":"99640dd9-8463-4c12-8282-b525b39fc10b",
      "Size":"21.47GB",
      "LBA_Data_Size":4096,
      "Namespace_Size":5242880
    }
  ]
}
```

Known issues

There are no known issues for the NVMe-oF host configuration for Ubuntu 24.04 with ONTAP release.

NVMe-oF host configuration for Ubuntu 24.04 with ONTAP

NVMe over Fabrics (NVMe-oF), including NVMe over Fibre Channel (NVMe/FC) and other transports, is supported with Ubuntu 22.04 with Asymmetric Namespace Access (ANA). In NVMe-oF environments, ANA is the equivalent of ALUA multipathing in iSCSI and FC environments and is implemented with in-kernel NVMe multipath.

The following support is available for the NVMe-oF host configuration for Ubuntu 22.04 with ONTAP:

- The NetApp plug-in in the native nvme-cli package displays ONTAP details for NVMe/FC namespaces.
- Use of NVMe and SCSI co-existent traffic on the same host on a given host bus adapter (HBA), without the explicit dm-multipath settings to prevent claiming NVMe namespaces.

For additional details on supported configurations, see the Interoperability Matrix Tool.

Features

Ubuntu 22.04 has in-kernel NVMe multipath enabled for NVMe namespaces by default. Therefore, there is no need for explicit settings.

Known limitations

SAN booting using the NVMe-oF protocol is currently not supported.

Validate software versions

You can use the following procedure to validate the minimum supported Ubuntu 22.04 software versions.

Steps

1. Install Ubuntu 22.04 on the server. After the installation is complete, verify that you are running the specified Ubuntu 22.04 kernel:

```
# uname -r
```

Example output:

```
5.15.0-101-generic
```

2. Install the `nvme-cli` package:

```
# apt list | grep nvme
```

Example output:

```
nvme-cli/jammy-updates,now 1.16-3ubuntu0.1 amd64
```

3. On the Ubuntu 22.04 host, check the hostnqn string at `/etc/nvme/hostnqn`:

```
# cat /etc/nvme/hostnqn
```

Example output

```
nqn.2014-08.org.nvmexpress:uuid:063a9fa0-438a-4737-b9b4-95a21c66d041
```

4. Verify that the `hostnqn` string matches the `hostnqn` string for the corresponding subsystem on the ONTAP array:

```
::> vserver nvme subsystem host show -vserver vs_106_fc_nvme
```

Example output:

```
Vserver     Subsystem          Host NQN
----------- ---------------
------------------------------------------------------------
vs_106_fc_nvme ub_106 nqn.2014-08.org.nvmexpress:uuid:c04702c8-e91e-
4353-9995-ba4536214631
```

If the `hostnqn` strings do not match, use the `vserver modify` command to update the `hostnqn` string on your corresponding ONTAP array subsystem to match the `hostnqn` string from `/etc/nvme/hostnqn` on the host.

Configure NVMe/FC

You can configure NVMe/FC for Broadcom/Emulex or Marvell/Qlogic adapters.

Broadcom/Emulex

1. Verify that you are using the supported adapter model.

```
# cat /sys/class/scsi_host/host*/modelname
```

Example output:

```
LPe36002-M64
LPe36002-M64
```

```
# cat /sys/class/scsi_host/host*/modeldesc
```

Example output:

```
Emulex LPe36002-M64 2-Port 64Gb Fibre Channel Adapter
Emulex LPe36002-M64 2-Port 64Gb Fibre Channel Adapter
```

2. Verify that you are using the recommended Broadcom `lpfc` firmware and inbox driver.

```
# cat /sys/class/scsi_host/host*/fwrev

14.2.673.40, sli-4:6:d
14.2.673.40, sli-4:6:d

# cat /sys/module/lpfc/version
0: 14.0.0.4
```

For the current list of supported adapter driver and firmware versions, see the Interoperability Matrix Tool.

3. Verify that `lpfc_enable_fc4_type` is set to `3`:

```
# cat /sys/module/lpfc/parameters/lpfc_enable_fc4_type
3
```

4. Verify that the initiator ports are up and running and that you can see the target LIFs:

```
# cat /sys/class/fc_host/host*/port_name
0x100000109bf0447c
0x100000109bf0447b
# cat /sys/class/fc_host/host*/port_state
Online
Online
# cat /sys/class/scsi_host/host*/nvme_info
        NVME Initiator Enabled
XRI Dist lpfc1 Total 6144 IO 5894 ELS 250
NVME LPORT lpfc1 WWPN x100000109bf0447c WWNN x200000109bf0447c DID
x022300 ONLINE
NVME RPORT       WWPN x200cd039eaa8138b WWNN x200ad039eaa8138b DID
x021509 TARGET DISCSRVC ONLINE
NVME RPORT       WWPN x2010d039eaa8138b WWNN x200ad039eaa8138b DID
x021108 TARGET DISCSRVC ONLINE

NVME Statistics
LS: Xmt 000000000e Cmpl 000000000e Abort 00000000
LS XMIT: Err 00000000  CMPL: xb 00000000 Err 00000000
Total FCP Cmpl 0000000000005238 Issue 000000000000523a OutIO
0000000000000002
        abort 00000000 noxri 00000000 nondlp 00000000 qdepth
00000000 wqerr 00000000 err 00000000
FCP CMPL: xb 00000000 Err 00000000

NVME Initiator Enabled
XRI Dist lpfc0 Total 6144 IO 5894 ELS 250
NVME LPORT lpfc0 WWPN x100000109bf0447b WWNN x200000109bf0447b DID
x022600 ONLINE
NVME RPORT       WWPN x200bd039eaa8138b WWNN x200ad039eaa8138b DID
x021409 TARGET DISCSRVC ONLINE
NVME RPORT       WWPN x200fd039eaa8138b WWNN x200ad039eaa8138b DID
x021008 TARGET DISCSRVC ONLINE

NVME Statistics
LS: Xmt 000000000e Cmpl 000000000e Abort 00000000
LS XMIT: Err 00000000  CMPL: xb 00000000 Err 00000000
Total FCP Cmpl 000000000000523c Issue 000000000000523e OutIO
0000000000000002
        abort 00000000 noxri 00000000 nondlp 00000000 qdepth
00000000 wqerr 00000000 err 00000000
FCP CMPL: xb 00000000 Err 00000000
```

Marvell/QLogic FC Adapter for NVMe/FC

The native inbox qla2xxx driver included in the Ubuntu 22.04 GA kernel has the latest upstream fixes.

These fixes are essential for ONTAP support.

1. Verify that you are running the supported adapter driver and firmware versions:

```
# cat /sys/class/fc_host/host*/symbolic_name
```

Example output

```
QLE2872 FW: v9.14.02 DVR: v10.02.06.200-k
QLE2872 FW: v9.14.02 DVR: v10.02.06.200-k
```

2. Verify that `ql2xnvmeenable` is set. This enables the Marvell adapter to function as an NVMe/FC initiator:

```
# cat /sys/module/qla2xxx/parameters/ql2xnvmeenable
1
```

Enable 1MB I/O (Optional)

ONTAP reports a Max Data Transfer Size (MDTS) of 8 in the Identify Controller data. This means the maximum I/O request size can be up to 1MB. To issue I/O requests of size 1MB for a Broadcom NVMe/FC host, you should increase the `lpfc` value of the `lpfc_sg_seg_cnt` parameter to 256 from the default value of 64.

These steps don't apply to Qlogic NVMe/FC hosts.

Steps

1. Set the `lpfc_sg_seg_cnt` parameter to 256:

```
cat /etc/modprobe.d/lpfc.conf
```

You should see an output similar to the following example:

```
options lpfc lpfc_sg_seg_cnt=256
```

2. Run the `dracut -f` command, and reboot the host.
3. Verify that the value for `lpfc_sg_seg_cnt` is 256:

```
cat /sys/module/lpfc/parameters/lpfc_sg_seg_cnt
```

Configure NVMe/TCP

NVMe/TCP does not have auto-connect functionality. Therefore, if a path goes down and is not reinstated within the default time out period of 10 minutes, NVMe/TCP cannot automatically reconnect. To prevent a time out, you should set the retry period for failover events to at least 30 minutes.

Steps

1. Verify that the initiator port can fetch the discovery log page data across the supported NVMe/TCP LIFs:

```
nvme discover -t tcp -w host-traddr -a traddr
```

Example output:

```
# nvme discover -t tcp -w 10.10.11.47-a 10.10.10.122

Discovery Log Number of Records 8, Generation counter 10
=====Discovery Log Entry 0======
trtype:  tcp
adrfam:  ipv4
subtype: current discovery subsystem
treq:    not specified
portid:  0
trsvcid: 8009
subnqn:  nqn.1992-
08.com.netapp:sn.bbfb4ee8dfb611edbd07d039ea165590:discovery
traddr:  10.10.10.122
eflags:  explicit discovery connections, duplicate discovery information
sectype: none
=====Discovery Log Entry 1======
trtype:  tcp
adrfam:  ipv4
subtype: current discovery subsystem
treq:    not specified
portid:  1
trsvcid: 8009
subnqn:  nqn.1992
08.com.netapp:sn.bbfb4ee8dfb611edbd07d039ea165590:discovery
traddr:  10.10.10.124
eflags:  explicit discovery connections, duplicate discovery information
sectype: none
=====Discovery Log Entry 2======
trtype:  tcp
```

2. Verify that the other NVMe/TCP initiator-target LIF combinations are able to successfully fetch discovery log page data:

```
nvme discover -t tcp -w host-traddr -a traddr
```

Example output:

```
#nvme discover -t tcp -w 10.10.10.47 -a 10.10.10.122
#nvme discover -t tcp -w 10.10.10.47 -a 10.10.10.124
#nvme discover -t tcp -w 10.10.11.47 -a 10.10.11.122
#nvme discover -t tcp -w 10.10.11.47 -a 10.10.11.
```

3. Run the nvme connect-all command across all the supported NVMe/TCP initiator-target LIFs across the nodes, and set the controller loss timeout period for at least 30 minutes or 1800 seconds:

```
nvme connect-all -t tcp -w host-traddr -a traddr -l 1800
```

Example output:

```
#   nvme    connect-all -t  tcp -w  10.10.10.47 -a  10.10.10.122 -l 1800
#   nvme    connect-all -t  tcp -w  10.10.10.47 -a  10.10.10.124 -l 1800
#   nvme    connect-all -t  tcp -w  10.10.11.47 -a  10.10.11.122 -l 1800
#   nvme    connect-all -t  tcp -w  10.10.11.47 -a  10.10.11.124 -l 1800
```

Validate NVMe-oF

You can use the following procedure to validate NVME-oF.

Steps

1. Verify that the in-kernel NVMe multipath is enabled:

```
# cat /sys/module/nvme_core/parameters/multipath
Y
```

2. Verify that the appropriate NVMe-oF settings (such as, model set to NetApp ONTAP Controller and load balancing iopolicy set to round-robin) for the respective ONTAP namespaces correctly reflect on the host:

```
# cat /sys/class/nvme-subsystem/nvme-subsys*/model
NetApp ONTAP Controller
NetApp ONTAP Controller
```

```
# cat /sys/class/nvme-subsystem/nvme-subsys*/iopolicy
round-robin
round-robin
```

3. Verify that the namespaces are created and correctly discovered on the host:

```
# nvme list
```

Example output:

```
Node         SN                   Model
------------------------------------------------------------
/dev/nvme0n1 81CZ5BQuUNfGAAAAAAAB   NetApp ONTAP Controller


Namespace Usage    Format             FW             Rev
------------------------------------------------------------
1                 21.47 GB / 21.47 GB   4 KiB + 0 B   FFFFFFFF
```

4. Verify that the controller state of each path is live and has the correct ANA status:

NVMe/FC

```
# nvme list-subsys /dev/nvme0n1
```

Example output:

```
nvme-subsys4 - NQN=nqn.1992-
08.com.netapp:sn.8763d311b2ac11ed950ed039ea951c46:subsystem. ub_106
\
+- nvme1 fc traddr=nn-0x20a6d039ea954d17:pn-
0x20a7d039ea954d17,host_traddr=nn-0x200000109b1b95ef:pn-
0x100000109b1b95ef live optimized
+- nvme2 fc traddr=nn-0x20a6d039ea954d17:pn-
0x20a8d039ea954d17,host_traddr=nn-0x200000109b1b95f0:pn-
0x100000109b1b95f0 live optimized
+- nvme3 fc traddr=nn-0x20a6d039ea954d17:pn-
0x20aad039ea954d17,host_traddr=nn-0x200000109b1b95f0:pn-
0x100000109b1b95f0 live non-optimized
+- nvme5 fc traddr=nn-0x20a6d039ea954d17:pn-
0x20a9d039ea954d17,host_traddr=nn-0x200000109b1b95ef:pn-
0x100000109b1b95ef live non-optimized
```

NVME/TCP

```
# nvme list-subsys /dev/nvme1n1
```

Example output:

```
nvme-subsys1 - NQN=nqn.1992- 08.com.netapp:sn.
bbfb4ee8dfb611edbd07d039ea165590:subsystem.rhel_tcp_95
+- nvme1 tcp
traddr=10.10.10.122,trsvcid=4420,host_traddr=10.10.10.47,src_addr=10
.10.10.47 live
+- nvme2 tcp
traddr=10.10.10.124,trsvcid=4420,host_traddr=10.10.10.47,src_addr=10
.10.10.47 live
+- nvme3 tcp
traddr=10.10.11.122,trsvcid=4420,host_traddr=10.10.11.47,src_addr=10
.10.11.47 live
+- nvme4 tcp
traddr=10.10.11.124,trsvcid=4420,host_traddr=10.10.11.47,src_addr=10
.10.11.47 live
```

5. Verify that the NetApp plug-in displays the correct values for each ONTAP namespace device:

Column

```
# nvme netapp ontapdevices -o column
```

Example output:

```
Device        Vserver   Namespace Path
--------------------- ------------------------------
/dev/nvme0n1 co_iscsi_tcp_ubuntu /vol/vol1/ns1



NSID       UUID                                   Size
-------------------------------------------------------------
1          79c2c569-b7fa-42d5-b870-d9d6d7e5fa84 21.47GB
```

JSON

```
# nvme netapp ontapdevices -o json
```

Example output

```
{

"ONTAPdevices" : [
{

"Device" : "/dev/nvme0n1",
"Vserver" : "co_iscsi_tcp_ubuntu",
"Namespace_Path" : "/vol/nvmevol1/ns1",
"NSID" : 1,
"UUID" : "79c2c569-b7fa-42d5-b870-d9d6d7e5fa84",
"Size" : "21.47GB",
"LBA_Data_Size" : 4096,
"Namespace_Size" : 5242880
},

]
}
```

Known issues

The NVMe-oF host configuration for Ubuntu 22.04 with ONTAP release has the following known issue:

NetApp Bug ID	Title	Description
CONTAPEXT-2037	Ubuntu 22.04 NVMe-oF hosts create duplicate Persistent Discovery Controllers	On NVMe-oF hosts, you can use the "nvme discover -p" command to create Persistent Discovery Controllers (PDCs). This command should only create one PDC for each initiator-target combination. However, if you are running Ubuntu 22.04 on an NVMe-oF host, a duplicate PDC is created each time "nvme discover -p" is executed. This leads to unnecessary usage of resources on both the host and the target.

Windows

Configure Windows Server 2025 with NVMe/FC for ONTAP

You can configure NVMe over Fibre Channel (NVMe/FC) on hosts running Windows Server 2025 for operation with ONTAP LUNs.

About this task

You can use the following support with the NVMe/FC host configuration for Windows 2025. You should also review the known limitations before starting the configuration process.

- Support available:

  Beginning with ONTAP 9.10.1, NVMe/FC is supported for Windows Server 2025.

  For a list of supported FC adapters and controllers, see the Hardware Universe. For the current list of supported configurations and versions, see the Interoperability Matrix Tool.

- Known limitations:

  Windows Failover Cluster isn't supported with NVMe/FC because ONTAP doesn't currently support persistent reservations with NVMe/FC.

> Broadcom ships an external driver for Windows NVMe/FC that is a translational SCSI NVMe driver and not a true NVMe/FC driver. The translational overhead doesn't necessarily impact performance, but it does negate the performance benefits of NVMe/FC. As a result, NVMe/FC and FCP performance is the same on Windows servers, unlike other operating systems such as Linux, where NVMe/FC performance is significantly better than that of FCP.

Enable NVMe/FC

Enable FC/NVMe on the Windows initiator host.

Steps

1. Install the Emulex HBA Manager utility on the Windows host.
2. On each of the HBA initiator ports, set the following HBA driver parameters:
   - EnableNVMe = 1

- NVMEMode = 0

3. Reboot the host.

Configure the Broadcom FC adapter

The Broadcom initiator can serve both NVMe/FC and FCP traffic through the same 32G FC adapter ports. For FCP and FC/NVMe, you should use the Microsft device-specific module (DSM) as the Microsoft Multipath I/O (MPIO) option.

A `hostnqn` is associated with each host bus adapter (HBA) port for the Broadcom adapter with FC/NVMe in a Windows environment. The `hostnqn` is formatted as shown in the following example:

```
nqn.2017-01.com.broadcom:ecd:nvmf:fc:100000109b1b9765
nqn.2017-01.com.broadcom:ecd:nvmf:fc:100000109b1b9766
```

Enable MPIO for NVMe devices

You need to enable MPIO for NVMe devices to complete the NVMe configuration on the Windows host.

Steps

1. Install Windows Host Utility Kit 7.1 to set the driver parameters that are common to both FC and NVMe.
2. Open the MPIO properties.
3. From the **Discover Multi-Paths** tab, add the device ID listed for NVMe.

   MPIO becomes aware of the NVMe devices, which are visible under disk management.

4. Open **Disk Management** and go to **Disk Properties**.
5. From the **MPIO** tab, select **Details**.
6. Set the following Microsoft DSM settings:
   - PathVerifiedPeriod: **10**
   - PathVerifyEnabled: **Enable**
   - RetryCount: **6**
   - RetryInterval: **1**
   - PDORemovedPeriod: **130**
7. Select the MPIO Policy **Round Robin with Subset**.
8. Change the registry values:

```
HKLM\SYSTEM\CurrentControlSet\Services\mpio\Parameters\PathRecoveryInter
val DWORD -> 30

HKLM\SYSTEM\CurrentControlSet\Services\mpio \Parameters\
UseCustomPathRecoveryInterval  DWORD-> 1
```

9. Reboot the host.

Validate the NVMe/FC configuration

Verify that the NVMe subsystems have been discovered and the ONTAP namespaces are correct for the NVMe-oF configuration.

Steps

1. Verify that "Port Type" is `FC+NVMe`:

   `listhba`

   Show example

   ```
   Port WWN       : 10:00:00:10:9b:1b:97:65
   Node WWN       : 20:00:00:10:9b:1b:97:65
   Fabric Name    : 10:00:c4:f5:7c:a5:32:e0
   Flags          : 8000e300
   Host Name      : INTEROP-57-159
   Mfg            : Emulex Corporation
   Serial No.     : FC71367217
   Port Number    : 0
   Mode           : Initiator
   PCI Bus Number : 94
   PCI Function   : 0
   Port Type      : FC+NVMe
   Model          : LPe32002-M2

   Port WWN       : 10:00:00:10:9b:1b:97:66
   Node WWN       : 20:00:00:10:9b:1b:97:66
   Fabric Name    : 10:00:c4:f5:7c:a5:32:e0
   Flags          : 8000e300
   Host Name      : INTEROP-57-159
   Mfg            : Emulex Corporation
   Serial No.     : FC71367217
   Port Number    : 1
   Mode           : Initiator
   PCI Bus Number : 94
   PCI Function   : 1
   Port Type      : FC+NVMe
   Model          : LPe32002-M2
   ```

2. Verify that the NVMe/FC subsystems have been discovered:

   ◦ `nvme-list`

Show example

```
NVMe Qualified Name     :  nqn.1992-
08.com.netapp:sn.a3b74c32db2911eab229d039ea141105:subsystem.win_n
vme_interop-57-159
Port WWN                :  20:09:d0:39:ea:14:11:04
Node WWN                :  20:05:d0:39:ea:14:11:04
Controller ID           :  0x0180
Model Number            :  NetApp ONTAP Controller
Serial Number           :  81CGZBPU5T/uAAAAAAAB
Firmware Version        :  FFFFFFFF
Total Capacity          :  Not Available
Unallocated Capacity    :  Not Available

NVMe Qualified Name     :  nqn.1992-
08.com.netapp:sn.a3b74c32db2911eab229d039ea141105:subsystem.win_n
vme_interop-57-159
Port WWN                :  20:06:d0:39:ea:14:11:04
Node WWN                :  20:05:d0:39:ea:14:11:04
Controller ID           :  0x0181
Model Number            :  NetApp ONTAP Controller
Serial Number           :  81CGZBPU5T/uAAAAAAAB
Firmware Version        :  FFFFFFFF
Total Capacity          :  Not Available
Unallocated Capacity    :  Not Available
Note: At present Namespace Management is not supported by NetApp
Arrays.
```

- `nvme-list`

Show example

```
NVMe Qualified Name     :  nqn.1992-
08.com.netapp:sn.a3b74c32db2911eab229d039ea141105:subsystem.win_n
vme_interop-57-159
Port WWN                :  20:07:d0:39:ea:14:11:04
Node WWN                :  20:05:d0:39:ea:14:11:04
Controller ID           :  0x0140
Model Number            :  NetApp ONTAP Controller
Serial Number           :  81CGZBPU5T/uAAAAAAAB
Firmware Version        :  FFFFFFFF
Total Capacity          :  Not Available
Unallocated Capacity    :  Not Available

NVMe Qualified Name     :  nqn.1992-
08.com.netapp:sn.a3b74c32db2911eab229d039ea141105:subsystem.win_n
vme_interop-57-159
Port WWN                :  20:08:d0:39:ea:14:11:04
Node WWN                :  20:05:d0:39:ea:14:11:04
Controller ID           :  0x0141
Model Number            :  NetApp ONTAP Controller
Serial Number           :  81CGZBPU5T/uAAAAAAAB
Firmware Version        :  FFFFFFFF
Total Capacity          :  Not Available
Unallocated Capacity    :  Not Available

Note: At present Namespace Management is not supported by NetApp
Arrays.
```

3. Verify that the namespaces have been created:

   `nvme-list-ns`

Show example

```
Active Namespaces (attached to controller 0x0141):

                                        SCSI           SCSI
SCSI
   NSID           DeviceName         Bus Number    Target Number
OS LUN
-----------  --------------------  ------------  ---------------
---------
0x00000001   \\.\PHYSICALDRIVE9          0                1
0
0x00000002   \\.\PHYSICALDRIVE10         0                1
1
0x00000003   \\.\PHYSICALDRIVE11         0                1
2
0x00000004   \\.\PHYSICALDRIVE12         0                1
3
0x00000005   \\.\PHYSICALDRIVE13         0                1
4
0x00000006   \\.\PHYSICALDRIVE14         0                1
5
0x00000007   \\.\PHYSICALDRIVE15         0                1
6
0x00000008   \\.\PHYSICALDRIVE16         0                1
7
```

Configure Windows Server 2022 with NVMe/FC for ONTAP

You can configure NVMe over Fibre Channel (NVMe/FC) on hosts running Windows Server 2022 for operation with ONTAP LUNs.

About this task

You can use the following support with the NVMe/FC host configuration for Windows 2022. You should also review the known limitations before starting the configuration process.

- Support available:

  Beginning with ONTAP 9.7, NVMe/FC is supported for Windows Server 2022.

  For a list of supported FC adapters and controllers, see the Hardware Universe. For the current list of supported configurations and versions, see the Interoperability Matrix Tool.

- Known limitations:

  Windows Failover Cluster isn't supported with NVMe/FC because ONTAP doesn't currently support

persistent reservations with NVMe/FC.

> Broadcom ships an external driver for Windows NVMe/FC that is a translational SCSI NVMe driver and not a true NVMe/FC driver. The translational overhead doesn't necessarily impact performance, but it does negate the performance benefits of NVMe/FC. As a result, NVMe/FC and FCP performance is the same on Windows servers, unlike other operating systems such as Linux, where NVMe/FC performance is significantly better than that of FCP.

Enable NVMe/FC

Enable FC/NVMe on the Windows initiator host.

Steps

1. Install the Emulex HBA Manager utility on the Windows host.
2. On each of the HBA initiator ports, set the following HBA driver parameters:
   - EnableNVMe = 1
   - NVMEMode = 0
3. Reboot the host.

Configure the Broadcom FC adapter

The Broadcom initiator can serve both NVMe/FC and FCP traffic through the same 32G FC adapter ports. For FCP and FC/NVMe, you should use the Microsft device-specific module (DSM) as the Microsoft Multipath I/O (MPIO) option.

A `hostnqn` is associated with each host bus adapter (HBA) port for the Broadcom adapter with FC/NVMe in a Windows environment. The `hostnqn` is formatted as shown in the following example:

```
nqn.2017-01.com.broadcom:ecd:nvmf:fc:100000109b1b9765
nqn.2017-01.com.broadcom:ecd:nvmf:fc:100000109b1b9766
```

Enable MPIO for NVMe devices

You need to enable MPIO for NVMe devices to complete the NVMe configuration on the Windows host.

Steps

1. Install Windows Host Utility Kit 7.1 to set the driver parameters that are common to both FC and NVMe.
2. Open the MPIO properties.
3. From the **Discover Multi-Paths** tab, add the device ID listed for NVMe.

   MPIO becomes aware of the NVMe devices, which are visible under disk management.
4. Open **Disk Management** and go to **Disk Properties**.
5. From the **MPIO** tab, select **Details**.
6. Set the following Microsoft DSM settings:
   - PathVerifiedPeriod: **10**

- PathVerifyEnabled: **Enable**
- RetryCount: **6**
- RetryInterval: **1**
- PDORemovedPeriod: **130**

7. Select the MPIO Policy **Round Robin with Subset**.
8. Change the registry values:

```
HKLM\SYSTEM\CurrentControlSet\Services\mpio\Parameters\PathRecoveryInter
val DWORD -> 30

HKLM\SYSTEM\CurrentControlSet\Services\mpio \Parameters\
UseCustomPathRecoveryInterval  DWORD-> 1
```

9. Reboot the host.

Validate NVMe/FC

Verify that the NVMe subsystems have been discovered and the ONTAP namespaces are correct for the NVMe-oF configuration.

Steps

1. Verify that "Port Type" is `FC+NVMe`:

```
listhba
```

Show example

```
Port WWN       : 10:00:00:10:9b:1b:97:65
Node WWN       : 20:00:00:10:9b:1b:97:65
Fabric Name    : 10:00:c4:f5:7c:a5:32:e0
Flags          : 8000e300
Host Name      : INTEROP-57-159
Mfg            : Emulex Corporation
Serial No.     : FC71367217
Port Number    : 0
Mode           : Initiator
PCI Bus Number : 94
PCI Function   : 0
Port Type      : FC+NVMe
Model          : LPe32002-M2

Port WWN       : 10:00:00:10:9b:1b:97:66
Node WWN       : 20:00:00:10:9b:1b:97:66
Fabric Name    : 10:00:c4:f5:7c:a5:32:e0
Flags          : 8000e300
Host Name      : INTEROP-57-159
Mfg            : Emulex Corporation
Serial No.     : FC71367217
Port Number    : 1
Mode           : Initiator
PCI Bus Number : 94
PCI Function   : 1
Port Type      : FC+NVMe
Model          : LPe32002-M2
```

2. Verify that the NVMe/FC subsystems have been discovered:
   - `nvme-list`

Show example

```
NVMe Qualified Name     :  nqn.1992-
08.com.netapp:sn.a3b74c32db2911eab229d039ea141105:subsystem.win_n
vme_interop-57-159
Port WWN                :  20:09:d0:39:ea:14:11:04
Node WWN                :  20:05:d0:39:ea:14:11:04
Controller ID           :  0x0180
Model Number            :  NetApp ONTAP Controller
Serial Number           :  81CGZBPU5T/uAAAAAAAB
Firmware Version        :  FFFFFFFF
Total Capacity          :  Not Available
Unallocated Capacity    :  Not Available

NVMe Qualified Name     :  nqn.1992-
08.com.netapp:sn.a3b74c32db2911eab229d039ea141105:subsystem.win_n
vme_interop-57-159
Port WWN                :  20:06:d0:39:ea:14:11:04
Node WWN                :  20:05:d0:39:ea:14:11:04
Controller ID           :  0x0181
Model Number            :  NetApp ONTAP Controller
Serial Number           :  81CGZBPU5T/uAAAAAAAB
Firmware Version        :  FFFFFFFF
Total Capacity          :  Not Available
Unallocated Capacity    :  Not Available
Note: At present Namespace Management is not supported by NetApp
Arrays.
```

- `nvme-list`

Show example

```
NVMe Qualified Name     :  nqn.1992-
08.com.netapp:sn.a3b74c32db2911eab229d039ea141105:subsystem.win_n
vme_interop-57-159
Port WWN                :  20:07:d0:39:ea:14:11:04
Node WWN                :  20:05:d0:39:ea:14:11:04
Controller ID           :  0x0140
Model Number            :  NetApp ONTAP Controller
Serial Number           :  81CGZBPU5T/uAAAAAAAB
Firmware Version        :  FFFFFFFF
Total Capacity          :  Not Available
Unallocated Capacity    :  Not Available

NVMe Qualified Name     :  nqn.1992-
08.com.netapp:sn.a3b74c32db2911eab229d039ea141105:subsystem.win_n
vme_interop-57-159
Port WWN                :  20:08:d0:39:ea:14:11:04
Node WWN                :  20:05:d0:39:ea:14:11:04
Controller ID           :  0x0141
Model Number            :  NetApp ONTAP Controller
Serial Number           :  81CGZBPU5T/uAAAAAAAB
Firmware Version        :  FFFFFFFF
Total Capacity          :  Not Available
Unallocated Capacity    :  Not Available

Note: At present Namespace Management is not supported by NetApp
Arrays.
```

3. Verify that the namespaces have been created:

   `nvme-list-ns`

Show example

```
Active Namespaces (attached to controller 0x0141):

                                          SCSI           SCSI
SCSI
   NSID           DeviceName         Bus Number    Target Number
OS LUN
-----------  --------------------  ------------  ---------------
---------
0x00000001   \\.\PHYSICALDRIVE9          0                1
0
0x00000002   \\.\PHYSICALDRIVE10         0                1
1
0x00000003   \\.\PHYSICALDRIVE11         0                1
2
0x00000004   \\.\PHYSICALDRIVE12         0                1
3
0x00000005   \\.\PHYSICALDRIVE13         0                1
4
0x00000006   \\.\PHYSICALDRIVE14         0                1
5
0x00000007   \\.\PHYSICALDRIVE15         0                1
6
0x00000008   \\.\PHYSICALDRIVE16         0                1
7
```

NVMe/FC Host Configuration for Windows Server 2019 with ONTAP

You can configure NVMe over Fibre Channel (NVMe/FC) on hosts running Windows Server 2019 for operation with ONTAP LUNs..

About this task

You can use the following support with the NVMe/FC host configuration for Windows 2019. You should also review the known limitations before starting the configuration process.

> ⓘ You can use the configuration settings provided in this procedure to configure cloud clients connected to Cloud Volumes ONTAP and Amazon FSx for ONTAP.

- Support available:

  Beginning with ONTAP 9.7, NVMe/FC is supported for Windows Server 2019.

  For a list of supported FC adapters and controllers, see the Hardware Universe. For the current list of supported configurations and versions, see the Interoperability Matrix Tool.

- Known limitations:

  Windows Failover Cluster isn't supported with NVMe/FC because ONTAP doesn't currently support persistent reservations with NVMe/FC.

> Broadcom ships an external driver for Windows NVMe/FC that is a translational SCSI NVMe driver and not a true NVMe/FC driver. The translational overhead doesn't necessarily impact performance, but it does negate the performance benefits of NVMe/FC. As a result, NVMe/FC and FCP performance is the same on Windows servers, unlike other operating systems such as Linux, where NVMe/FC performance is significantly better than that of FCP.

Enable NVMe/FC

Enable FC/NVMe on the Windows initiator host.

Steps

1. Install the Emulex HBA Manager utility on the Windows host.
2. On each of the HBA initiator ports, set the following HBA driver parameters:
   - EnableNVMe = 1
   - NVMEMode = 0
3. Reboot the host.

Configure the Broadcom FC adapter

The Broadcom initiator can serve both NVMe/FC and FCP traffic through the same 32G FC adapter ports. For FCP and FC/NVMe, you should use the Microsft device-specific module (DSM) as the Microsoft Multipath I/O (MPIO) option.

A `hostnqn` is associated with each host bus adapter (HBA) port for the Broadcom adapter with FC/NVMe in a Windows environment. The `hostnqn` is formatted as shown in the following example:

```
nqn.2017-01.com.broadcom:ecd:nvmf:fc:100000109b1b9765
nqn.2017-01.com.broadcom:ecd:nvmf:fc:100000109b1b9766
```

Enable MPIO for NVMe devices

You need to enable MPIO for NVMe devices to complete the NVMe configuration on the Windows host.

Steps

1. Install Windows Host Utility Kit 7.1 to set the driver parameters that are common to both FC and NVMe.
2. Open the MPIO properties.
3. From the **Discover Multi-Paths** tab, add the device ID listed for NVMe.

   MPIO becomes aware of the NVMe devices, which are visible under disk management.

4. Open **Disk Management** and go to **Disk Properties**.
5. From the **MPIO** tab, select **Details**.

6. Set the following Microsoft DSM settings:
   - PathVerifiedPeriod: **10**
   - PathVerifyEnabled: **Enable**
   - RetryCount: **6**
   - RetryInterval: **1**
   - PDORemovedPeriod: **130**
7. Select the MPIO Policy **Round Robin with Subset**.
8. Change the registry values:

```
HKLM\SYSTEM\CurrentControlSet\Services\mpio\Parameters\PathRecoveryInter
val DWORD -> 30

HKLM\SYSTEM\CurrentControlSet\Services\mpio \Parameters\
UseCustomPathRecoveryInterval  DWORD-> 1
```

9. Reboot the host.

Validate NVMe/FC

Verify that the NVMe subsystems have been discovered and the ONTAP namespaces are correct for the NVMe-oF configuration.

Steps

1. Verify that "Port Type" is `FC+NVMe`:

```
listhba
```

Show example

```
Port WWN       : 10:00:00:10:9b:1b:97:65
Node WWN       : 20:00:00:10:9b:1b:97:65
Fabric Name    : 10:00:c4:f5:7c:a5:32:e0
Flags          : 8000e300
Host Name      : INTEROP-57-159
Mfg            : Emulex Corporation
Serial No.     : FC71367217
Port Number    : 0
Mode           : Initiator
PCI Bus Number : 94
PCI Function   : 0
Port Type      : FC+NVMe
Model          : LPe32002-M2

Port WWN       : 10:00:00:10:9b:1b:97:66
Node WWN       : 20:00:00:10:9b:1b:97:66
Fabric Name    : 10:00:c4:f5:7c:a5:32:e0
Flags          : 8000e300
Host Name      : INTEROP-57-159
Mfg            : Emulex Corporation
Serial No.     : FC71367217
Port Number    : 1
Mode           : Initiator
PCI Bus Number : 94
PCI Function   : 1
Port Type      : FC+NVMe
Model          : LPe32002-M2
```

2. Verify that the NVMe/FC subsystems have been discovered:
   - `nvme-list`

Show example

```
NVMe Qualified Name     :  nqn.1992-
08.com.netapp:sn.a3b74c32db2911eab229d039ea141105:subsystem.win_n
vme_interop-57-159
Port WWN                :  20:09:d0:39:ea:14:11:04
Node WWN                :  20:05:d0:39:ea:14:11:04
Controller ID           :  0x0180
Model Number            :  NetApp ONTAP Controller
Serial Number           :  81CGZBPU5T/uAAAAAAAB
Firmware Version        :  FFFFFFFF
Total Capacity          :  Not Available
Unallocated Capacity    :  Not Available

NVMe Qualified Name     :  nqn.1992-
08.com.netapp:sn.a3b74c32db2911eab229d039ea141105:subsystem.win_n
vme_interop-57-159
Port WWN                :  20:06:d0:39:ea:14:11:04
Node WWN                :  20:05:d0:39:ea:14:11:04
Controller ID           :  0x0181
Model Number            :  NetApp ONTAP Controller
Serial Number           :  81CGZBPU5T/uAAAAAAAB
Firmware Version        :  FFFFFFFF
Total Capacity          :  Not Available
Unallocated Capacity    :  Not Available
Note: At present Namespace Management is not supported by NetApp
Arrays.
```

- `nvme-list`

Show example

```
NVMe Qualified Name     :  nqn.1992-
08.com.netapp:sn.a3b74c32db2911eab229d039ea141105:subsystem.win_n
vme_interop-57-159
Port WWN                :  20:07:d0:39:ea:14:11:04
Node WWN                :  20:05:d0:39:ea:14:11:04
Controller ID           :  0x0140
Model Number            :  NetApp ONTAP Controller
Serial Number           :  81CGZBPU5T/uAAAAAAAB
Firmware Version        :  FFFFFFFF
Total Capacity          :  Not Available
Unallocated Capacity    :  Not Available

NVMe Qualified Name     :  nqn.1992-
08.com.netapp:sn.a3b74c32db2911eab229d039ea141105:subsystem.win_n
vme_interop-57-159
Port WWN                :  20:08:d0:39:ea:14:11:04
Node WWN                :  20:05:d0:39:ea:14:11:04
Controller ID           :  0x0141
Model Number            :  NetApp ONTAP Controller
Serial Number           :  81CGZBPU5T/uAAAAAAAB
Firmware Version        :  FFFFFFFF
Total Capacity          :  Not Available
Unallocated Capacity    :  Not Available

Note: At present Namespace Management is not supported by NetApp
Arrays.
```

3. Verify that the namespaces have been created:

   `nvme-list-ns`

Show example

```
Active Namespaces (attached to controller 0x0141):

                                          SCSI           SCSI
SCSI
   NSID           DeviceName         Bus Number    Target Number
OS LUN
-----------  --------------------  ------------  ---------------
---------
0x00000001   \\.\PHYSICALDRIVE9          0                1
0
0x00000002   \\.\PHYSICALDRIVE10         0                1
1
0x00000003   \\.\PHYSICALDRIVE11         0                1
2
0x00000004   \\.\PHYSICALDRIVE12         0                1
3
0x00000005   \\.\PHYSICALDRIVE13         0                1
4
0x00000006   \\.\PHYSICALDRIVE14         0                1
5
0x00000007   \\.\PHYSICALDRIVE15         0                1
6
0x00000008   \\.\PHYSICALDRIVE16         0                1
7
```

Configure Windows Server 2016 with NVMe/FC for ONTAP

You can configure NVMe over Fibre Channel (NVMe/FC) on hosts running Windows Server 2016 for operation with ONTAP LUNs.

About this task

You can use the following support with the NVMe/FC host configuration for Windows 2016. You should also review the known limitations before starting the configuration process.

> You can use the configuration settings provided in this procedure to configure cloud clients connected to Cloud Volumes ONTAP and Amazon FSx for ONTAP.

- Support available:

  Beginning with ONTAP 9.7, NVMe/FC is supported for Windows Server 2016.

  For a list of supported FC adapters and controllers, see the Hardware Universe. For the current list of supported configurations and versions, see the Interoperability Matrix Tool.

- Known limitations:

  Windows Failover Cluster isn't supported with NVMe/FC because ONTAP doesn't currently support persistent reservations with NVMe/FC.

> Broadcom ships an external driver for Windows NVMe/FC that is a translational SCSI NVMe driver and not a true NVMe/FC driver. The translational overhead doesn't necessarily impact performance, but it does negate the performance benefits of NVMe/FC. As a result, NVMe/FC and FCP performance is the same on Windows servers, unlike other operating systems such as Linux, where NVMe/FC performance is significantly better than that of FCP.

Enable NVMe/FC

Enable FC/NVMe on the Windows initiator host.

Steps

1. Install the Emulex HBA Manager utility on the Windows host.
2. On each of the HBA initiator ports, set the following HBA driver parameters:
   - EnableNVMe = 1
   - NVMEMode = 0
3. Reboot the host.

Configure the Broadcom FC adapter

The Broadcom initiator can serve both NVMe/FC and FCP traffic through the same 32G FC adapter ports. For FCP and FC/NVMe, you should use the Microsft device-specific module (DSM) as the Microsoft Multipath I/O (MPIO) option.

A `hostnqn` is associated with each host bus adapter (HBA) port for the Broadcom adapter with FC/NVMe in a Windows environment. The `hostnqn` is formatted as shown in the following example:

```
nqn.2017-01.com.broadcom:ecd:nvmf:fc:100000109b1b9765
nqn.2017-01.com.broadcom:ecd:nvmf:fc:100000109b1b9766
```

Enable MPIO for NVMe devices

You need to enable MPIO for NVMe devices to complete the NVMe configuration on the Windows host.

Steps

1. Install Windows Host Utility Kit 7.1 to set the driver parameters that are common to both FC and NVMe.
2. Open the MPIO properties.
3. From the **Discover Multi-Paths** tab, add the device ID listed for NVMe.

   MPIO becomes aware of the NVMe devices, which are visible under disk management.

4. Open **Disk Management** and go to **Disk Properties**.
5. From the **MPIO** tab, select **Details**.

6. Set the following Microsoft DSM settings:
   - PathVerifiedPeriod: **10**
   - PathVerifyEnabled: **Enable**
   - RetryCount: **6**
   - RetryInterval: **1**
   - PDORemovedPeriod: **130**
7. Select the MPIO Policy **Round Robin with Subset**.
8. Change the registry values:

```
HKLM\SYSTEM\CurrentControlSet\Services\mpio\Parameters\PathRecoveryInter
val DWORD -> 30

HKLM\SYSTEM\CurrentControlSet\Services\mpio \Parameters\
UseCustomPathRecoveryInterval  DWORD-> 1
```

9. Reboot the host.

Validate NVMe/FC

Verify that the NVMe subsystems have been discovered and the ONTAP namespaces are correct for the NVMe-oF configuration.

Steps

1. Verify that "Port Type" is `FC+NVMe`:

```
listhba
```

Show example

```
Port WWN       : 10:00:00:10:9b:1b:97:65
Node WWN       : 20:00:00:10:9b:1b:97:65
Fabric Name    : 10:00:c4:f5:7c:a5:32:e0
Flags          : 8000e300
Host Name      : INTEROP-57-159
Mfg            : Emulex Corporation
Serial No.     : FC71367217
Port Number    : 0
Mode           : Initiator
PCI Bus Number : 94
PCI Function   : 0
Port Type      : FC+NVMe
Model          : LPe32002-M2

Port WWN       : 10:00:00:10:9b:1b:97:66
Node WWN       : 20:00:00:10:9b:1b:97:66
Fabric Name    : 10:00:c4:f5:7c:a5:32:e0
Flags          : 8000e300
Host Name      : INTEROP-57-159
Mfg            : Emulex Corporation
Serial No.     : FC71367217
Port Number    : 1
Mode           : Initiator
PCI Bus Number : 94
PCI Function   : 1
Port Type      : FC+NVMe
Model          : LPe32002-M2
```

2. Verify that the NVMe/FC subsystems have been discovered:
   - `nvme-list`

Show example

```
NVMe Qualified Name     :  nqn.1992-
08.com.netapp:sn.a3b74c32db2911eab229d039ea141105:subsystem.win_n
vme_interop-57-159
Port WWN                :  20:09:d0:39:ea:14:11:04
Node WWN                :  20:05:d0:39:ea:14:11:04
Controller ID           :  0x0180
Model Number            :  NetApp ONTAP Controller
Serial Number           :  81CGZBPU5T/uAAAAAAAB
Firmware Version        :  FFFFFFFF
Total Capacity          :  Not Available
Unallocated Capacity    :  Not Available

NVMe Qualified Name     :  nqn.1992-
08.com.netapp:sn.a3b74c32db2911eab229d039ea141105:subsystem.win_n
vme_interop-57-159
Port WWN                :  20:06:d0:39:ea:14:11:04
Node WWN                :  20:05:d0:39:ea:14:11:04
Controller ID           :  0x0181
Model Number            :  NetApp ONTAP Controller
Serial Number           :  81CGZBPU5T/uAAAAAAAB
Firmware Version        :  FFFFFFFF
Total Capacity          :  Not Available
Unallocated Capacity    :  Not Available
Note: At present Namespace Management is not supported by NetApp
Arrays.
```

- nvme-list

Show example

```
NVMe Qualified Name     :  nqn.1992-
08.com.netapp:sn.a3b74c32db2911eab229d039ea141105:subsystem.win_n
vme_interop-57-159
Port WWN                :  20:07:d0:39:ea:14:11:04
Node WWN                :  20:05:d0:39:ea:14:11:04
Controller ID           :  0x0140
Model Number            :  NetApp ONTAP Controller
Serial Number           :  81CGZBPU5T/uAAAAAAAB
Firmware Version        :  FFFFFFFF
Total Capacity          :  Not Available
Unallocated Capacity    :  Not Available

NVMe Qualified Name     :  nqn.1992-
08.com.netapp:sn.a3b74c32db2911eab229d039ea141105:subsystem.win_n
vme_interop-57-159
Port WWN                :  20:08:d0:39:ea:14:11:04
Node WWN                :  20:05:d0:39:ea:14:11:04
Controller ID           :  0x0141
Model Number            :  NetApp ONTAP Controller
Serial Number           :  81CGZBPU5T/uAAAAAAAB
Firmware Version        :  FFFFFFFF
Total Capacity          :  Not Available
Unallocated Capacity    :  Not Available

Note: At present Namespace Management is not supported by NetApp
Arrays.
```

3. Verify that the namespaces have been created:

```
nvme-list-ns
```

Show example

```
Active Namespaces (attached to controller 0x0141):

                                          SCSI           SCSI
SCSI
   NSID           DeviceName         Bus Number    Target Number
OS LUN
-----------  --------------------  ------------  ---------------
---------
0x00000001   \\.\PHYSICALDRIVE9          0               1
0
0x00000002   \\.\PHYSICALDRIVE10         0               1
1
0x00000003   \\.\PHYSICALDRIVE11         0               1
2
0x00000004   \\.\PHYSICALDRIVE12         0               1
3
0x00000005   \\.\PHYSICALDRIVE13         0               1
4
0x00000006   \\.\PHYSICALDRIVE14         0               1
5
0x00000007   \\.\PHYSICALDRIVE15         0               1
6
0x00000008   \\.\PHYSICALDRIVE16         0               1
7
```

Configure Windows Server 2012 R2 with NVMe/FC for ONTAP

You can configure NVMe over Fibre Channel (NVMe/FC) on hosts running Windows Server 2012 R2 for operation with ONTAP LUNs.

About this task

You can use the following support with the NVMe/FC host configuration for Windows 2012 R2. You should also review the known limitations before starting the configuration process.

> You can use the configuration settings provided in this procedure to configure cloud clients connected to Cloud Volumes ONTAP and Amazon FSx for ONTAP.

- Support available:

  Beginning with ONTAP 9.7, NVMe/FC is supported for Windows Server 2012 R2.

  For a list of supported FC adapters and controllers, see the Hardware Universe. For the current list of supported configurations and versions, see the Interoperability Matrix Tool.

- Known limitations:

  Windows Failover Cluster isn't supported with NVMe/FC because ONTAP doesn't currently support persistent reservations with NVMe/FC.

> Broadcom ships an external driver for Windows NVMe/FC that is a translational SCSI NVMe driver and not a true NVMe/FC driver. The translational overhead doesn't necessarily impact performance, but it does negate the performance benefits of NVMe/FC. As a result, NVMe/FC and FCP performance is the same on Windows servers, unlike other operating systems such as Linux, where NVMe/FC performance is significantly better than that of FCP.

Enable NVMe/FC

Enable FC/NVMe on the Windows initiator host.

Steps

1. Install the Emulex HBA Manager utility on the Windows host.
2. On each of the HBA initiator ports, set the following HBA driver parameters:
   - EnableNVMe = 1
   - NVMEMode = 0
3. Reboot the host.

Configure the Broadcom FC adapter

The Broadcom initiator can serve both NVMe/FC and FCP traffic through the same 32G FC adapter ports. For FCP and FC/NVMe, you should use the Microsft device-specific module (DSM) as the Microsoft Multipath I/O (MPIO) option.

A `hostnqn` is associated with each host bus adapter (HBA) port for the Broadcom adapter with FC/NVMe in a Windows environment. The `hostnqn` is formatted as shown in the following example:

```
nqn.2017-01.com.broadcom:ecd:nvmf:fc:100000109b1b9765
nqn.2017-01.com.broadcom:ecd:nvmf:fc:100000109b1b9766
```

Enable MPIO for NVMe devices

You need to enable MPIO for NVMe devices to complete the NVMe configuration on the Windows host.

Steps

1. Install Windows Host Utility Kit 7.1 to set the driver parameters that are common to both FC and NVMe.
2. Open the MPIO properties.
3. From the **Discover Multi-Paths** tab, add the device ID listed for NVMe.

   MPIO becomes aware of the NVMe devices, which are visible under disk management.

4. Open **Disk Management** and go to **Disk Properties**.
5. From the **MPIO** tab, select **Details**.

6. Set the following Microsoft DSM settings:
   - PathVerifiedPeriod: **10**
   - PathVerifyEnabled: **Enable**
   - RetryCount: **6**
   - RetryInterval: **1**
   - PDORemovedPeriod: **130**
7. Select the MPIO Policy **Round Robin with Subset**.
8. Change the registry values:

```
HKLM\SYSTEM\CurrentControlSet\Services\mpio\Parameters\PathRecoveryInter
val DWORD -> 30

HKLM\SYSTEM\CurrentControlSet\Services\mpio \Parameters\
UseCustomPathRecoveryInterval  DWORD-> 1
```

9. Reboot the host.

Validate NVMe/FC

Verify that the NVMe subsystems have been discovered and the ONTAP namespaces are correct for the NVMe-oF configuration.

Steps

1. Verify that "Port Type" is `FC+NVMe`:

```
listhba
```

Show example

```
Port WWN       : 10:00:00:10:9b:1b:97:65
Node WWN       : 20:00:00:10:9b:1b:97:65
Fabric Name    : 10:00:c4:f5:7c:a5:32:e0
Flags          : 8000e300
Host Name      : INTEROP-57-159
Mfg            : Emulex Corporation
Serial No.     : FC71367217
Port Number    : 0
Mode           : Initiator
PCI Bus Number : 94
PCI Function   : 0
Port Type      : FC+NVMe
Model          : LPe32002-M2

Port WWN       : 10:00:00:10:9b:1b:97:66
Node WWN       : 20:00:00:10:9b:1b:97:66
Fabric Name    : 10:00:c4:f5:7c:a5:32:e0
Flags          : 8000e300
Host Name      : INTEROP-57-159
Mfg            : Emulex Corporation
Serial No.     : FC71367217
Port Number    : 1
Mode           : Initiator
PCI Bus Number : 94
PCI Function   : 1
Port Type      : FC+NVMe
Model          : LPe32002-M2
```

2. Verify that the NVMe/FC subsystems have been discovered:
   - `nvme-list`

Show example

```
NVMe Qualified Name     :  nqn.1992-
08.com.netapp:sn.a3b74c32db2911eab229d039ea141105:subsystem.win_n
vme_interop-57-159
Port WWN                :  20:09:d0:39:ea:14:11:04
Node WWN                :  20:05:d0:39:ea:14:11:04
Controller ID           :  0x0180
Model Number            :  NetApp ONTAP Controller
Serial Number           :  81CGZBPU5T/uAAAAAAAB
Firmware Version        :  FFFFFFFF
Total Capacity          :  Not Available
Unallocated Capacity    :  Not Available

NVMe Qualified Name     :  nqn.1992-
08.com.netapp:sn.a3b74c32db2911eab229d039ea141105:subsystem.win_n
vme_interop-57-159
Port WWN                :  20:06:d0:39:ea:14:11:04
Node WWN                :  20:05:d0:39:ea:14:11:04
Controller ID           :  0x0181
Model Number            :  NetApp ONTAP Controller
Serial Number           :  81CGZBPU5T/uAAAAAAAB
Firmware Version        :  FFFFFFFF
Total Capacity          :  Not Available
Unallocated Capacity    :  Not Available
Note: At present Namespace Management is not supported by NetApp
Arrays.
```

- `nvme-list`

Show example

```
NVMe Qualified Name     :  nqn.1992-
08.com.netapp:sn.a3b74c32db2911eab229d039ea141105:subsystem.win_n
vme_interop-57-159
Port WWN                :  20:07:d0:39:ea:14:11:04
Node WWN                :  20:05:d0:39:ea:14:11:04
Controller ID           :  0x0140
Model Number            :  NetApp ONTAP Controller
Serial Number           :  81CGZBPU5T/uAAAAAAAB
Firmware Version        :  FFFFFFFF
Total Capacity          :  Not Available
Unallocated Capacity    :  Not Available

NVMe Qualified Name     :  nqn.1992-
08.com.netapp:sn.a3b74c32db2911eab229d039ea141105:subsystem.win_n
vme_interop-57-159
Port WWN                :  20:08:d0:39:ea:14:11:04
Node WWN                :  20:05:d0:39:ea:14:11:04
Controller ID           :  0x0141
Model Number            :  NetApp ONTAP Controller
Serial Number           :  81CGZBPU5T/uAAAAAAAB
Firmware Version        :  FFFFFFFF
Total Capacity          :  Not Available
Unallocated Capacity    :  Not Available

Note: At present Namespace Management is not supported by NetApp
Arrays.
```

3. Verify that the namespaces have been created:

   `nvme-list-ns`

Show example

```
Active Namespaces (attached to controller 0x0141):

                                        SCSI           SCSI
SCSI
   NSID           DeviceName         Bus Number    Target Number
OS LUN
-----------  --------------------  ------------  ---------------
---------
0x00000001   \\.\PHYSICALDRIVE9          0               1
0
0x00000002   \\.\PHYSICALDRIVE10         0               1
1
0x00000003   \\.\PHYSICALDRIVE11         0               1
2
0x00000004   \\.\PHYSICALDRIVE12         0               1
3
0x00000005   \\.\PHYSICALDRIVE13         0               1
4
0x00000006   \\.\PHYSICALDRIVE14         0               1
5
0x00000007   \\.\PHYSICALDRIVE15         0               1
6
0x00000008   \\.\PHYSICALDRIVE16         0               1
7
```

Troubleshoot NVMe-oF with ONTAP storage for Linux Host Utilities

Troubleshoot NVMe-oF failures for Oracle Linux, RHEL, Rocky Linux, and SUSE Linux Enterprise Server hosts with ONTAP storage.

Before you start troubleshooting, verify that you are running a configuration that is compliant to the Interoperability Matrix Tool specifications and then proceed with the next steps to debug any host side issues.

The troubleshooting instructions are not applicable to AIX, ESXi, Proxmox, and Windows hosts.

Enable verbose logging

If you have an issue with your configuration, verbose logging can provide essential information for troubleshooting.

Steps

Set verbose logging for LPFC or Qlogic (Qla2xxx).

LPFC

Set the lpfc driver for NVMe/FC.

1. Set the `lpfc_log_verbose` driver setting to any of the following values to log NVMe/FC events.

```
#define LOG_NVME 0x00100000 /* NVME general events. */
#define LOG_NVME_DISC 0x00200000 /* NVME Discovery/Connect events.
*/
#define LOG_NVME_ABTS 0x00400000 /* NVME ABTS events. */
#define LOG_NVME_IOERR 0x00800000 /* NVME IO Error events. */
```

2. After setting the values, run the `dracut-f` command and reboot the host.
3. Verify the settings:

```
# cat /etc/modprobe.d/lpfc.conf options lpfc
lpfc_log_verbose=0xf00083

# cat /sys/module/lpfc/parameters/lpfc_log_verbose 15728771
```

Qla2xxx

There is no specific qla2xxx logging for NVMe/FC similar to that for the `lpfc` driver. Instead, set the general qla2xxx logging level.

1. Append the `ql2xextended_error_logging=0x1e400000` value to the corresponding `modprobe qla2xxx conf` file.
2. Execute the `dracut -f` command and then reboot the host.
3. After reboot, verify that the verbose logging has been enabled:

```
# cat /etc/modprobe.d/qla2xxx.conf
```

You should see an output similar to the following example:

```
options qla2xxx ql2xnvmeenable=1
ql2xextended_error_logging=0x1e400000
# cat /sys/module/qla2xxx/parameters/ql2xextended_error_logging
507510784
```

Common "nvme-cli" errors and workarounds

The errors displayed by `nvme-cli` during `nvme discover`, `nvme connect`, or `nvme connect-all` operations and the workarounds are shown in the following table:

Error message	Probable cause	Workaround
`Failed to write to /dev/nvme-fabrics: Invalid argument`	Incorrect syntax	Verify that you are using the correct syntax for the `nvme discover`, `nvme connect`, and `nvme connect-all` commands.

Error message	Probable cause	Workaround

`Failed to write to /dev/nvme-fabrics: No such file or directory`	Multiple issues can trigger this, for example, providing wrong arguments to the NVMe commands is one of the common causes.	• Verify that you have passed the correct arguments (such as, correct WWNN string, WWPN string, and more) to the commands.
		• If the arguments are correct, but you still see this error, check whether the `/sys/class/scsi_host/host*/nvme_info` command output is correct, the NVMe initiator is displayed as `Enabled`, and the NVMe/FC target LIFs are correctly displayed under the remote ports sections. Example:
		`# cat /sys/class/scsi_host/host*/nvme_info`
		`NVME Initiator Enabled`
		`NVME LPORT lpfc0 WWPN x10000090fae0ec9d WWNN x20000090fae0ec9d DID x012000 ONLINE`
		`NVME RPORT WWPN x200b00a098c80f09 WWNN x200a00a098c80f09 DID x010601 TARGET DISCSRVC ONLINE`
		`NVME Statistics`
		`LS: Xmt 0000000000000006 Cmpl 0000000000000006`
		`FCP: Rd 0000000000000071 Wr 0000000000000005 IO 0000000000000031 Cmpl 00000000000000a6 Outstanding 0000000000000001`
		`NVME Initiator Enabled`
		`NVME LPORT lpfc1 WWPN x10000090fae0ec9e WWNN x20000090fae0ec9e DID x012400 ONLINE`
		`NVME RPORT WWPN x200900a098c80f09 WWNN x200800a098c80f09 DID x010301 TARGET DISCSRVC ONLINE`
		`NVME Statistics`
		`LS: Xmt 0000000000000006 Cmpl 0000000000000006`
		`FCP: Rd 0000000000000073 Wr 0000000000000005 IO 0000000000000031 Cmpl 00000000000000a8 Outstanding 0000000000000001`
		• If the `nvme_info` command output doesn't display the target LIFs as shown in the example above, check the `/var/log/messages` and `dmesg` command outputs for any suspicious NVMe/FC failures, and report or fix accordingly.

Error message	Probable cause	Workaround
`No discovery log entries to fetch`	Generally observed when the `/etc/nvme/hostnqn` string has not been added to the corresponding subsystem on the NetApp array or an incorrect `hostnqn` string has been added to the respective subsystem.	Verify that the exact `/etc/nvme/hostnqn` string is added to the corresponding subsystem on the NetApp array (verify using the `vserver nvme subsystem host show` command).
`Failed to write to /dev/nvme-fabrics: Operation already in progress`	Observed when the controller associations or specified operation is already created or in the process of being created. This could happen as part of the auto-connect scripts installed above.	None. Try running the `nvme discover` command again after some time. For `nvme connect` and `connect-all`, run the `nvme list` command to verify that the namespace devices are already created and displayed on the host.

When to contact technical support

If you continue to have issues, collect the following files and command outputs and contact NetApp support for further triage:

```
cat /sys/class/scsi_host/host*/nvme_info
/var/log/messages
dmesg
nvme discover output as in:
nvme discover --transport=fc --traddr=nn-0x200a00a098c80f09:pn
-0x200b00a098c80f09 --host-traddr=nn-0x20000090fae0ec9d:pn
-0x10000090fae0ec9d
nvme list
nvme list-subsys /dev/nvmeXnY
```

Legal notices

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

Copyright

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

Trademarks

NETAPP, the NETAPP logo, and the marks listed on the NetApp Trademarks page are trademarks of NetApp, Inc. Other company and product names may be trademarks of their respective owners.

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

Patents

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

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

Privacy policy

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

- Notice for AIX
- Notice for Linux
- Notice for Solaris
- Notice for Windows

Copyright information

Copyright © 2026 NetApp, Inc. All Rights Reserved. Printed in the U.S. No part of this document covered by copyright may be reproduced in any form or by any means—graphic, electronic, or mechanical, including photocopying, recording, taping, or storage in an electronic retrieval system—without prior written permission of the copyright owner.

Software derived from copyrighted NetApp material is subject to the following license and disclaimer:

THIS SOFTWARE IS PROVIDED BY NETAPP “AS IS” AND WITHOUT ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE, WHICH ARE HEREBY DISCLAIMED. IN NO EVENT SHALL NETAPP BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

NetApp reserves the right to change any products described herein at any time, and without notice. NetApp assumes no responsibility or liability arising from the use of products described herein, except as expressly agreed to in writing by NetApp. The use or purchase of this product does not convey a license under any patent rights, trademark rights, or any other intellectual property rights of NetApp.

The product described in this manual may be protected by one or more U.S. patents, foreign patents, or pending applications.

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

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

Trademark information

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