NVMe over RoCE Setup

E-Series Systems

NetApp
December 14, 2022

Table of Contents

NVMe over RoCE Setup

Verify Linux support and review restrictions

As a first step, you should verify that your Linux configuration is supported and also review the controller, switch, host, and recovery restrictions.

Verify the Linux configuration is supported

To ensure reliable operation, you create an implementation plan and then use the NetApp Interoperability Matrix Tool (IMT) to verify that the entire configuration is supported.

Steps

1. Go to the NetApp Interoperability Matrix Tool.
2. Click on the **Solution Search** tile.
3. In the **Protocols › SAN Host** area, click the **Add** button next to **E-Series SAN Host**.
4. Click **View Refine Search Criteria**.

   The Refine Search Criteria section is displayed. In this section you may select the protocol that applies, as well as other criteria for the configuration such as Operating System, NetApp OS, and Host Multipath driver.

5. Select the criteria you know you want for your configuration, and then see what compatible configuration elements apply.
6. As necessary, make the updates for your operating system and protocol that are prescribed in the tool.

   Detailed information for your chosen configuration is accessible on the View Supported Configurations page by clicking the right page arrow.

Verify NVMe over RoCE restrictions

Before using NVMe over RoCE, see the NetApp Interoperability Matrix Tool to review the latest controller, host, and recovery restrictions.

Switch restrictions

RISK OF DATA LOSS. You must enable flow control for use with Global Pause Control on the switch to eliminate the risk of data loss in an NVMe over RoCE environment.

Storage and disaster recovery restrictions

- Asynchronous and synchronous mirroring are not supported.
- Thin provisioning (the creation of thin volumes) is not supported.

Configure IP addresses using DHCP

To configure communications between the management station and the storage array,

use Dynamic Host Configuration Protocol (DHCP) to provide IP addresses.

What you'll need

A DHCP server installed and configured on the same subnet as the storage management ports.

About this task

Each storage array has either one controller (simplex) or two controllers (duplex), and each controller has two storage management ports. Each management port will be assigned an IP address.

The following instructions refer to a storage array with two controllers (a duplex configuration).

Steps

1. If you have not already done so, connect an Ethernet cable to the management station and to management port 1 on each controller (A and B).

   The DHCP server assigns an IP address to port 1 of each controller.

   > Do not use management port 2 on either controller. Port 2 is reserved for use by NetApp technical personnel.

   > If you disconnect and reconnect the Ethernet cable, or if the storage array is power-cycled, DHCP assigns IP addresses again. This process occurs until static IP addresses are configured. It is recommended that you avoid disconnecting the cable or power-cycling the array.

   If the storage array cannot get DHCP-assigned IP addresses within 30 seconds, the following default IP addresses are set:

   - Controller A, port 1: 169.254.128.101
   - Controller B, port 1: 169.254.128.102
   - Subnet mask: 255.255.0.0

2. Locate the MAC address label on the back of each controller, and then provide your network administrator with the MAC address for port 1 of each controller.

   Your network administrator needs the MAC addresses to determine the IP address for each controller. You will need the IP addresses to connect to your storage system through your browser.

Install SANtricity Storage Manager for SMcli (SANtricity software version 11.53 or earlier)

If you are using SANtricity software 11.53 or earlier, you can install the SANtricity Storage Manager software on your management station to help manage the array.

SANtricity Storage Manager includes the command line interface (CLI) for additional management tasks, and also the Host Context Agent for pushing host configuration information to the storage array controllers through the I/O path.

If you are using SANtricity software 11.60 and newer, you do not need to follow these steps. The SANtricity Secure CLI (SMcli) is included in the SANtricity OS and downloadable through the SANtricity System Manager. For more information on how to download the SMcli through the SANtricity System Manager, refer to the *Download command line interface (CLI)* topic under the SANtricity System Manager Online Help.

What you'll need

- SANtricity software 11.53 or earlier.
- Correct administrator or superuser privileges.
- A system for the SANtricity Storage Manager client with the following minimum requirements:
  - **RAM**: 2 GB for Java Runtime Engine
  - **Disk space**: 5 GB
  - **OS/Architecture**: For guidance on determining the supported operating system versions and architectures, go to NetApp Support. From the **Downloads** tab, go to **Downloads › E-Series SANtricity Storage Manager**.

About this task

This task describes how to install SANtricity Storage Manager on both the Windows and Linux OS platforms, because both Windows and Linux are common management station platforms when Linux is used for the data host.

Steps

1. Download the SANtricity software release at NetApp Support. From the **Downloads** tab, go to **Downloads › E-Series SANtricity Storage Manager**.
2. Run the SANtricity installer.

Windows	Linux
Double-click the SMIA*.exe installation package to start the installation.	a. Go to the directory where the SMIA*.bin installation package is located. b. If the temp mount point does not have execute permissions, set the `IATEMPDIR` variable. Example: `IATEMPDIR=/root ./SMIA-LINUXX64-11.25.0A00.0002.bin` c. Run the `chmod +x SMIA*.bin` command to grant execute permission to the file. d. Run the `./SMIA*.bin` command to start the installer.

3. Use the installation wizard to install the software on the management station.

Access SANtricity System Manager and use Setup wizard

To configure your storage array, you can use the Setup wizard in SANtricity System Manager.

SANtricity System Manager is a web-based interface embedded on each controller. To access the user interface, you point a browser to the controller's IP address. A setup wizard helps you get started with system configuration.

What you'll need

- Out-of-band management.
- A management station for accessing SANtricity System Manager that includes one of the following browsers:

Browser	Minimum version
Google Chrome	79
Microsoft Internet Explorer	11
Microsoft Edge	79
Mozilla Firefox	70
Safari	12

About this task

The wizard automatically relaunches when you open System Manager or refresh your browser and *at least one* of the following conditions is met:

- No pools and volume groups are detected.
- No workloads are detected.
- No notifications are configured.

Steps

1. From your browser, enter the following URL: `https://<DomainNameOrIPAddress>`

   `IPAddress` is the address for one of the storage array controllers.

   The first time SANtricity System Manager is opened on an array that has not been configured, the Set Administrator Password prompt appears. Role-based access management configures four local roles: admin, support, security, and monitor. The latter three roles have random passwords that cannot be guessed. After you set a password for the admin role, you can change all of the passwords using the admin credentials. For more information about the four local user roles, see the online help available in the SANtricity System Manager user interface.

2. Enter the System Manager password for the admin role in the Set Administrator Password and Confirm Password fields, and then click **Set Password**.

   The Setup wizard launches if there are no pools, volumes groups, workloads, or notifications configured.

3. Use the Setup wizard to perform the following tasks:
   - **Verify hardware (controllers and drives)** — Verify the number of controllers and drives in the storage array. Assign a name to the array.

- **Verify hosts and operating systems** — Verify the host and operating system types that the storage array can access.
- **Accept pools** — Accept the recommended pool configuration for the express installation method. A pool is a logical group of drives.
- **Configure alerts** — Allow System Manager to receive automatic notifications when a problem occurs with the storage array.
- **Enable AutoSupport** — Automatically monitor the health of your storage array and have dispatches sent to technical support.

4. If you have not already created a volume, create one by going to **Storage › Volumes › Create › Volume**.

   For more information, see the online help for SANtricity System Manager.

Configure the switch

You configure the switches according to the vendor's recommendations for NVMe over RoCE. These recommendations might include both configuration directives as well as code updates.

RISK OF DATA LOSS. You must enable flow control for use with Global Pause Control on the switch to eliminate the risk of data loss in an NVMe over RoCE environment.

Steps

1. Enable Ethernet pause frame flow control **end to end** as the best practice configuration.
2. Consult your network administrator for tips on selecting the best configuration for your environment.

Set up NVMe over RoCE on the host side

NVMe initiator configuration in a RoCE environment includes installing and configuring the rdma-core and nvme-cli packages, configuring initiator IP addresses, and setting up the NVMe-oF layer on the host.

What you'll need

You must be running the latest compatible RHEL 7, RHEL 8, and RHEL 9 SUSE Linux Enterprise Server 12 and 15 service pack operating system. See the NetApp Interoperability Matrix Tool for a complete list of the latest requirements.

Steps

1. Install the rdma and nvme-cli packages:

   SLES 12 or SLES 15

   ```
   # zypper install rdma-core
   # zypper install nvme-cli
   ```

   RHEL 7, RHEL 8, and RHEL 9

```
# yum install rdma-core
# yum install nvme-cli
```

2. Get the host NQN, which will be used to configure the host to an array.

```
# cat /etc/nvme/hostnqn
```

3. Set up IPv4 IP addresses on the ethernet ports used to connect NVMe over RoCE. For each network interface, create a configuration script that contains the different variables for that interface.

   The variables used in this step are based on server hardware and the network environment. The variables include the `IPADDR` and `GATEWAY`. These are example instructions for SLES and RHEL:

   SLES 12 and SLES 15

   Create the example file `/etc/sysconfig/network/ifcfg-eth4` with the following contents.

```
BOOTPROTO='static'
BROADCAST=
ETHTOOL_OPTIONS=
IPADDR='192.168.1.87/24'
GATEWAY='192.168.1.1'
MTU=
NAME='MT27800 Family [ConnectX-5]'
NETWORK=
REMOTE_IPADDR=
STARTMODE='auto'
```

   Then, create the example file `/etc/sysconfig/network/ifcfg-eth5`:

```
BOOTPROTO='static'
BROADCAST=
ETHTOOL_OPTIONS=
IPADDR='192.168.2.87/24'
GATEWAY='192.168.2.1'
MTU=
NAME='MT27800 Family [ConnectX-5]'
NETWORK=
REMOTE_IPADDR=
STARTMODE='auto'
```

   RHEL 7, RHEL 8, and RHEL 9

   Create the example file `/etc/sysconfig/network-scripts/ifcfg-eth4` with the following contents.

```
BOOTPROTO='static'
BROADCAST=
ETHTOOL_OPTIONS=
IPADDR='192.168.1.87/24’
GATEWAY='192.168.1.1'
MTU=
NAME='MT27800 Family [ConnectX-5]'
NETWORK=
REMOTE_IPADDR=
STARTMODE='auto'
```

Then, create the example file `/etc/sysconfig/network-scripts/ifcfg-eth5`:

```
BOOTPROTO='static'
BROADCAST=
ETHTOOL_OPTIONS=
IPADDR='192.168.2.87/24'
GATEWAY='192.168.2.1'
MTU=
NAME='MT27800 Family [ConnectX-5]'
NETWORK=
REMOTE_IPADDR=
STARTMODE='auto'
```

4. Enable the network interfaces:

```
# ifup eth4
# ifup eth5
```

5. Set up the NVMe-oF layer on the host. Create the following file under `/etc/modules-load.d/` to load the `nvme-rdma` kernel module and make sure the kernel module will always be on, even after a reboot:

```
# cat /etc/modules-load.d/nvme-rdma.conf
  nvme-rdma
```

To verify the `nvme-rdma` kernel module is loaded, run this command:

```
# lsmod | grep nvme
nvme_rdma               36864  0
nvme_fabrics            24576  1 nvme_rdma
nvme_core              114688  5 nvme_rdma,nvme_fabrics
rdma_cm                114688  7
rpcrdma,ib_srpt,ib_srp,nvme_rdma,ib_iser,ib_isert,rdma_ucm
ib_core                393216  15
rdma_cm,ib_ipoib,rpcrdma,ib_srpt,ib_srp,nvme_rdma,iw_cm,ib_iser,ib_umad,
ib_isert,rdma_ucm,ib_uverbs,mlx5_ib,qedr,ib_cm
t10_pi                  16384  2 sd_mod,nvme_core
```

Configure storage array NVMe over RoCE connections

If your controller includes a connection for NVMe over RoCE (RDMA over Converged Ethernet), you can configure the NVMe port settings from the Hardware page or the System page in SANtricity System Manager.

What you'll need

- An NVMe over RoCE host port on your controller; otherwise, the NVMe over RoCE settings are not available in System Manager.
- The IP address of the host connection.

About this task

You can access the NVMe over RoCE configuration from the **Hardware** page or from **Settings › System**. This task describes how to configure the ports from the Hardware page.

The NVMe over RoCE settings and functions appear only if your storage array's controller includes an NVMe over RoCE port.

Steps

1. From the System Manager interface, select **Hardware**.
2. Click the controller with the NVMe over RoCE port you want to configure.

   The controller's context menu appears.
3. Select **Configure NVMe over RoCE ports**.

   The **Configure NVMe over RoCE ports** dialog box opens.
4. In the drop-down list, select the port you want to configure, and then click **Next**.
5. Select the port configuration settings you want to use, and then click **Next**.

   To see all port settings, click the **Show more port settings** link on the right of the dialog box.

Port Setting	Description
Configured ethernet port speed	Select the desired speed. The options that appear in the drop-down list depend on the maximum speed that your network can support (for example, 10 Gbps). Possible values include: • Auto-negotiate • 10 Gbps • 25 Gbps • 40 Gbps • 50 Gbps • 100 Gbps • 200 Gbps When a 200Gb-capable HIC is attached with a QSFP56 cable, auto-negotiate is only available when you are connecting to Mellanox switches and/or adapters. The configured NVMe over RoCE port speed should match the speed capability of the SFP on the selected port. All ports must be set to the same speed.
Enable IPv4 and/or Enable IPv6	Select one or both options to enable support for IPv4 and IPv6 networks.
MTU size (Available by clicking **Show more port settings**.)	If necessary, enter a new size in bytes for the maximum transmission unit (MTU). The default MTU size is 1500 bytes per frame. You must enter a value between 1500 and 4200.

If you selected **Enable IPv4**, a dialog box opens for selecting IPv4 settings after you click **Next**. If you selected **Enable IPv6**, a dialog box opens for selecting IPv6 settings after you click **Next**. If you selected both options, the dialog box for IPv4 settings opens first, and then after you click **Next**, the dialog box for IPv6 settings opens.

6. Configure the IPv4 and/or IPv6 settings, either automatically or manually. To see all port settings, click the **Show more settings** link on the right of the dialog box.

Port setting	Description
Automatically obtain configuration from DHCP server	Select this option to obtain the configuration automatically.

Port setting	Description
Manually specify static configuration	Select this option, and then enter a static address in the fields. For IPv4, include the network subnet mask and gateway. For IPv6, include the routable IP addresses and router IP address. If there is only one routable IP address, set the remaining address to 0:0:0:0:0:0:0:0.
Enable VLAN support (Available by clicking **Show more settings**.)	This option is only available in an iSCSI environment. It is not available in an NVMe over RoCE environment.
Enable ethernet priority (Available by clicking **Show more settings**.)	This option is only available in an iSCSI environment. It is not available in an NVMe over RoCE environment.

7. Click **Finish**.

Discover and connect to the storage from the host

Before making definitions of each host in SANtricity System Manager, you must discover the target controller ports from the host, and then establish NVMe connections.

Steps

1. Discover available subsystems on the NVMe-oF target for all paths using the following command:

```
nvme discover -t rdma -a target_ip_address
```

In this command, `target_ip_address` is the IP address of the target port.

The `nvme discover` command discovers all controller ports in the subsystem, regardless of host access.

```
# nvme discover -t rdma -a 192.168.1.77
Discovery Log Number of Records 2, Generation counter 0
=====Discovery Log Entry 0======
trtype:  rdma
adrfam:  ipv4
subtype: nvme subsystem
treq:    not specified
portid:  0
trsvcid: 4420
subnqn:  nqn.1992-08.com.netapp:5700.600a098000a527a7000000005ab3af94
traddr:  192.168.1.77
rdma_prtype: roce
rdma_qptype: connected
rdma_cms:    rdma-cm
rdma_pkey: 0x0000
=====Discovery Log Entry 1======
trtype:  rdma
adrfam:  ipv4
subtype: nvme subsystem
treq:    not specified
portid:  1
trsvcid: 4420
subnqn:  nqn.1992-08.com.netapp:5700.600a098000a527a7000000005ab3af94
traddr:  192.168.2.77
rdma_prtype: roce
rdma_qptype: connected
rdma_cms:    rdma-cm
rdma_pkey: 0x0000
```

2. Repeat step 1 for any other connections.
3. Connect to the discovered subsystem on the first path using the command: `nvme connect -t rdma -n discovered_sub_nqn -a target_ip_address -Q queue_depth_setting -l controller_loss_timeout_period`

The command listed above does not persist through reboot. The `NVMe connect` command will need to be executed after each reboot to re-establish the NVMe connections.

Connections are not established for any discovered port inaccessible by the host.

If you specify a port number using this command, the connection fails. The default port is the only port set up for connections.

The recommended queue depth setting is 1024. Override the default setting of 128 with 1024 using the `-Q 1024`command line option, as shown in the following example.

The recommended controller loss timeout period in seconds is 60 minutes (3600 seconds). Override the default setting of 600 seconds with 3600 seconds using the `-l 3600` command line option, as shown in the following example.

```
# nvme connect -t rdma -a 192.168.1.77 -n nqn.1992-
08.com.netapp:5700.600a098000a527a7000000005ab3af94 -Q 1024 -l 3600
# nvme connect -t rdma -a 192.168.2.77 -n nqn.1992-
08.com.netapp:5700.600a098000a527a7000000005ab3af94 -Q 1024 -l 3600
```

4. Repeat step 3 to connect the discovered subsystem on the second path.

Define a host

Using SANtricity System Manager, you define the hosts that send data to the storage array. Defining a host is one of the steps required to let the storage array know which hosts are attached to it and to allow I/O access to the volumes.

About this task

Keep these guidelines in mind when you define a host:

- You must define the host identifier ports that are associated with the host.
- Make sure that you provide the same name as the host's assigned system name.
- This operation does not succeed if the name you choose is already in use.
- The length of the name cannot exceed 30 characters.

Steps

1. Select **Storage › Hosts**.
2. Click **Create › Host**.

   The Create Host dialog box appears.

3. Select the settings for the host as appropriate.

Setting	Description
Name	Type a name for the new host.
Host operating system type	Select one of the following options from the drop-down list: • **Linux** for SANtricity 11.60 and newer • **Linux DM-MP (Kernel 3.10 or later)** for pre-SANtricity 11.60

Setting	Description
Host interface type	Select the host interface type that you want to use. If the array you configure only has one available host interface type, this setting may not be available to select.
Host ports	Do one of the following: • **Select I/O Interface** If the host ports have logged in, you can select host port identifiers from the list. This is the recommended method. • **Manual add** If the host ports have not logged in, look at /etc/nvme/hostnqn on the host to find the hostnqn identifiers and associate them with the host definition. You can manually enter the host port identifiers or copy/paste them from the /etc/nvme/hostnqn file (one at a time) into the **Host ports** field. You must add one host port identifier at a time to associate it with the host, but you can continue to select as many identifiers that are associated with the host. Each identifier is displayed in the **Host ports** field. If necessary, you also can remove an identifier by selecting the **X** next to it.

4. Click **Create**.

Result

After the host is successfully created, SANtricity System Manager creates a default name for each host port configured for the host.

The default alias is `<Hostname_Port Number>`. For example, the default alias for the first port created for `host IPT is IPT_1`.

Assign a volume

You must assign a volume (namespace) to a host or host cluster so it can be used for I/O operations. This assignment grants a host or host cluster access to one or more namespaces in a storage array.

About this task

Keep these guidelines in mind when you assign volumes:

- You can assign a volume to only one host or host cluster at a time.
- Assigned volumes are shared between controllers in the storage array.
- The same namespace ID (NSID) cannot be used twice by a host or a host cluster to access a volume. You must use a unique NSID.

Assigning a volume fails under these conditions:

- All volumes are assigned.
- The volume is already assigned to another host or host cluster.

The ability to assign a volume is unavailable under these conditions:

- No valid hosts or host clusters exist.
- All volume assignments have been defined.

All unassigned volumes are displayed, but functions for hosts with or without Data Assurance (DA) apply as follows:

- For a DA-capable host, you can select volumes that are either DA-enabled or not DA-enabled.
- For a host that is not DA-capable, if you select a volume that is DA-enabled, a warning states that the system must automatically turn off DA on the volume before assigning the volume to the host.

Steps

1. Select **Storage › Hosts**.
2. Select the host or host cluster to which you want to assign volumes, and then click **Assign Volumes**.

   A dialog box appears that lists all the volumes that can be assigned. You can sort any of the columns or type something in the **Filter** box to make it easier to find particular volumes.
3. Select the checkbox next to each volume that you want to assign or select the checkbox in the table header to select all volumes.
4. Click **Assign** to complete the operation.

Result

After successfully assigning a volume or volumes to a host or a host cluster, the system performs the following actions:

- The assigned volume receives the next available NSID. The host uses the NSID to access the volume.
- The user-supplied volume name appears in volume listings associated to the host.

Display the volumes visible to the host

You can use the SMdevices tool to view volumes currently visible on the host. This tool is part of the nvme-cli package, and can be used as an alternative to the `nvme list` command.

To view information about each NVMe path to an E-Series volume, use the `nvme netapp smdevices [-o <format>]` command. The output <format> can be normal (the default if -o is not used), column, or json.

```
# nvme netapp smdevices
/dev/nvme1n1, Array Name ICTM0706SYS04, Volume Name NVMe2, NSID 1, Volume
ID 000015bd5903df4a00a0980000af4462, Controller A, Access State unknown,
2.15GB
/dev/nvme1n2, Array Name ICTM0706SYS04, Volume Name NVMe3, NSID 2, Volume
ID 000015c05903e24000a0980000af4462, Controller A, Access State unknown,
2.15GB
/dev/nvme1n3, Array Name ICTM0706SYS04, Volume Name NVMe4, NSID 4, Volume
ID 00001bb0593a46f400a0980000af4462, Controller A, Access State unknown,
2.15GB
/dev/nvme1n4, Array Name ICTM0706SYS04, Volume Name NVMe6, NSID 6, Volume
ID 00001696593b424b00a0980000af4112, Controller A, Access State unknown,
2.15GB
/dev/nvme2n1, Array Name ICTM0706SYS04, Volume Name NVMe2, NSID 1, Volume
ID 000015bd5903df4a00a0980000af4462, Controller B, Access State unknown,
2.15GB
/dev/nvme2n2, Array Name ICTM0706SYS04, Volume Name NVMe3, NSID 2, Volume
ID 000015c05903e24000a0980000af4462, Controller B, Access State unknown,
2.15GB
/dev/nvme2n3, Array Name ICTM0706SYS04, Volume Name NVMe4, NSID 4, Volume
ID 00001bb0593a46f400a0980000af4462, Controller B, Access State unknown,
2.15GB
/dev/nvme2n4, Array Name ICTM0706SYS04, Volume Name NVMe6, NSID 6, Volume
ID 00001696593b424b00a0980000af4112, Controller B, Access State unknown,
2.15GB
```

Set up failover on the host

To provide a redundant path to the storage array, you can configure the host to run failover.

What you'll need

You must install the required packages on your system.

- For Red Hat (RHEL) hosts, verify the packages are installed by running `rpm -q device-mapper-multipath`
- For SLES hosts, verify the packages are installed by running `rpm -q multipath-tools`

> Refer to the NetApp Interoperability Matrix Tool to ensure any required updates are installed, as multipathing might not work correctly with the GA versions of SLES or RHEL.

About this task

RHEL 7 and SLES 12 use Device Mapper Multipath (DMMP) for multipathing for NVMe over RoCE. RHEL 8, RHEL 9, and SLES 15 use a built-in Native NVMe Failover. Depending on which OS you are running, some additional configuration of multipath is required to get it running properly.

Enable Device Mapper Multipath (DMMP) for RHEL 7 or SLES 12

By default, DM-MP is disabled in RHEL and SLES. Complete the following steps to enable DM-MP components on the host.

Steps

1. Add the NVMe E-Series device entry to the devices section of the `/etc/multipath.conf` file, as shown in the following example:

```
devices {
        device {
                vendor "NVME"
                product "NetApp E-Series*"
                path_grouping_policy group_by_prio
                failback immediate
                no_path_retry 30
        }
}
```

2. Configure `multipathd` to start at system boot.

```
# systemctl enable multipathd
```

3. Start `multipathd` if it is not currently running.

```
# systemctl start multipathd
```

4. Verify the status of `multipathd` to make sure it is active and running:

```
# systemctl status multipathd
```

Set up RHEL 8 with Native NVMe Multipathing

Native NVMe Multipathing is disabled by default in RHEL 8 and must be enabled using the following procedure.

1. Set up the `modprobe` rule to turn on Native NVMe Multipathing.

```
# echo "options nvme_core multipath=y" >> /etc/modprobe.d/50-
nvme_core.conf
```

2. Remake `initramfs` with the new `modprobe` parameter.

```
# dracut -f
```

3. Reboot the server to bring it up with the Native NVMe Multipathing enabled.

```
# reboot
```

4. Verify that Native NVMe Multipathing is enabled after the host boots back up.

```
# cat /sys/module/nvme_core/parameters/multipath
```

   a. If the command output is `N`, then Native NVMe Multipathing is still disabled.

   b. If the command output is `Y`, then Native NVMe Multipathing is enabled and any NVMe devices you discover will use it.

For RHEL 9 and SLES 15, Native NVMe Multipathing is enabled by default and no additional configuration is required.

Access NVMe volumes for virtual device targets

You can configure the I/O that is directed to the device target based on which OS (and by extension multipathing method) you are using.

For RHEL 7 and SLES 12, I/O is directed to virtual device targets by the Linux host. DM-MP manages the physical paths underlying these virtual targets.

Virtual devices are I/O targets

Make sure you are running I/O only to the virtual devices created by DM-MP and not to the physical device paths. If you are running I/O to the physical paths, DM-MP cannot manage a failover event and the I/O fails.

You can access these block devices through the `dm` device or the `symlink` in /dev/mapper. For example:

```
/dev/dm-1
/dev/mapper/eui.00001bc7593b7f5f00a0980000af4462
```

Example

The following example output from the `nvme list` command shows the host node name and its correlation with the namespace ID.

```
NODE         SN           MODEL            NAMESPACE

/dev/nvme1n1 021648023072 NetApp E-Series 10
/dev/nvme1n2 021648023072 NetApp E-Series 11
/dev/nvme1n3 021648023072 NetApp E-Series 12
/dev/nvme1n4 021648023072 NetApp E-Series 13
/dev/nvme2n1 021648023151 NetApp E-Series 10
/dev/nvme2n2 021648023151 NetApp E-Series 11
/dev/nvme2n3 021648023151 NetApp E-Series 12
/dev/nvme2n4 021648023151 NetApp E-Series 13
```

Column	Description
`Node`	The node name includes two parts: • The notation `nvme1` represents controller A and `nvme2` represents controller B. • The notation `n1`, `n2`, and so on represent the namespace identifier from the host perspective. These identifiers are repeated in the table, once for controller A and once for controller B.
`Namespace`	The Namespace column lists the namespace ID (NSID), which is the identifier from the storage array perspective.

In the following `multipath -ll` output, the optimized paths are shown with a `prio` value of 50, while the non-optimized paths are shown with a `prio` value of 10.

The Linux operating system routes I/O to the path group that is shown as `status=active`, while the path groups listed as `status=enabled` are available for failover.

```
eui.00001bc7593b7f500a0980000af4462 dm-0 NVME,NetApp E-Series
size=15G features='1 queue_if_no_path' hwhandler='0' wp=rw
|-+- policy='service-time 0' prio=50 status=active
| `- #:#:#:# nvme1n1 259:5 active ready running
`-+- policy='service-time 0' prio=10 status=enabled
  `- #:#:#:# nvme2n1 259:9  active ready running

eui.00001bc7593b7f5f00a0980000af4462 dm-0 NVME,NetApp E-Series
size=15G features='1 queue_if_no_path' hwhandler='0' wp=rw
|-+- policy='service-time 0' prio=0 status=enabled
| `- #:#:#:# nvme1n1 259:5 failed faulty running
`-+- policy='service-time 0' prio=10 status=active
  `- #:#:#:# nvme2n1 259:9  active ready running
```

Line item	Description
`policy='service-time 0' prio=50 status=active`	This line and the following line show that `nvme1n1`, which is the namespace with an NSID of 10, is optimized on the path with a `prio` value of 50 and a `status` value of `active`. This namespace is owned by controller A.
`policy='service-time 0' prio=10 status=enabled`	This line shows the failover path for namespace 10, with a `prio` value of 10 and a `status` value of `enabled`. I/O is not being directed to the namespace on this path at the moment. This namespace is owned by controller B.
`policy='service-time 0' prio=0 status=enabled`	This example shows `multipath -ll`output from a different point in time, while controller A is rebooting. The path to namespace 10 is shown as `failed faulty running` with a `prio` value of 0 and a `status` value of `enabled`.
`policy='service-time 0' prio=10 status=active`	Note that the `active` path refers to `nvme2`, so the I/O is being directed on this path to controller B.

Accessing NVMe volumes for physical NVMe device targets

You can configure the I/O directed to the device target based on which OS (and by extension multipathing method) you are using.

For RHEL 8, RHEL 9, and SLES 15, I/O is directed to the physical NVMe device targets by the Linux host. A native NVMe multipathing solution manages the physical paths underlying the single apparent physical device displayed by the host.

Physical NVMe devices are I/O targets

It is best practice to run I/O to the links in `/dev/disk/by-id/nvme-eui.[uuid#]` rather than directly to the physical nvme device path `/dev/nvme[subsys#]n[id#]`. The link between these two locations can be found using the following command:

```
# ls /dev/disk/by-id/ -l
lrwxrwxrwx 1 root root 13 Oct 18 15:14 nvme-
eui.0000320f5cad32cf00a0980000af4112 -> ../../nvme0n1
```

I/O run to `/dev/disk/by-id/nvme-eui.[uuid#]` will be passed directly through `/dev/nvme[subsys#]n[id#]` which has all paths virtualized underneath it using the Native NVMe multipathing solution.

You can view your paths by running:

```
# nvme list-subsys
```

Example output:

```
nvme-subsys0 - NQN=nqn.1992-
08.com.netapp:5700.600a098000a522500000000589aa8a6
\
+- nvme0 rdma traddr=192.4.21.131 trsvcid=4420 live
+- nvme1 rdma traddr=192.4.22.141 trsvcid=4420 live
```

If you specify a namespace device when using the `nvme list-subsys` command, it provides additional information about the paths to that namespace:

```
# nvme list-subsys /dev/nvme0n1
nvme-subsys0 - NQN=nqn.1992-
08.com.netapp:5700.600a098000af44620000000058d5dd96
\
 +- nvme0 rdma traddr=192.168.130.101 trsvcid=4420 live non-optimized
 +- nvme1 rdma traddr=192.168.131.101 trsvcid=4420 live non-optimized
 +- nvme2 rdma traddr=192.168.130.102 trsvcid=4420 live optimized
 +- nvme3 rdma traddr=192.168.131.102 trsvcid=4420 live optimized
```

There are also hooks into the multipath commands to allow you to view your path information for native failover through them as well:

```
#multipath -ll
```

To view the path information, the following must be set in /etc/multipath.conf:

```
defaults {
        enable_foreign nvme
}
```

Example output:

```
eui.0000a0335c05d57a00a0980000a5229d [nvme]:nvme0n9 NVMe,Netapp E-
Series,08520001
size=4194304 features='n/a' hwhandler='ANA' wp=rw
|-+- policy='n/a' prio=50 status=optimized
| `- 0:0:1 nvme0c0n1 0:0 n/a optimized    live
`-+- policy='n/a' prio-10 status=non-optimized
`- 0:1:1 nvme0c1n1 0:0 n/a non-optimized    live
```

Create filesystems (RHEL 7 and SLES 12)

For RHEL 7 and SLES 12, you create a file system on the namespace and mount the filesystem.

Steps

1. Run the `multipath -ll` command to get a list of `/dev/mapper/dm` devices.

   ```
   # multipath -ll
   ```

   The result of this command shows two devices, `dm-19` and `dm-16`:

   ```
   eui.00001ffe5a94ff8500a0980000af4444 dm-19 NVME,NetApp E-Series
   size=10G features='1 queue_if_no_path' hwhandler='0' wp=rw
   |-+- policy='service-time 0' prio=50 status=active
   | |- #:#:#:# nvme0n19 259:19  active ready running
   | `- #:#:#:# nvme1n19 259:115 active ready running
   `-+- policy='service-time 0' prio=10 status=enabled
     |- #:#:#:# nvme2n19 259:51  active ready running
     `- #:#:#:# nvme3n19 259:83  active ready running
   eui.00001fd25a94fef000a0980000af4444 dm-16 NVME,NetApp E-Series
   size=16G features='1 queue_if_no_path' hwhandler='0' wp=rw
   |-+- policy='service-time 0' prio=50 status=active
   | |- #:#:#:# nvme0n16 259:16  active ready running
   | `- #:#:#:# nvme1n16 259:112 active ready running
   `-+- policy='service-time 0' prio=10 status=enabled
     |- #:#:#:# nvme2n16 259:48  active ready running
     `- #:#:#:# nvme3n16 259:80  active ready running
   ```

2. Create a file system on the partition for each `/dev/mapper/eui-` device.

   The method for creating a file system varies depending on the file system chosen. This example shows creating an `ext4` file system.

```
# mkfs.ext4 /dev/mapper/dm-19
mke2fs 1.42.11 (09-Jul-2014)
Creating filesystem with 2620928 4k blocks and 655360 inodes
Filesystem UUID: 97f987e9-47b8-47f7-b434-bf3ebbe826d0
Superblock backups stored on blocks:
        32768, 98304, 163840, 229376, 294912, 819200, 884736, 1605632

Allocating group tables: done
Writing inode tables: done
Creating journal (32768 blocks): done
Writing superblocks and filesystem accounting information: done
```

3. Create a folder to mount the new device.

```
# mkdir /mnt/ext4
```

4. Mount the device.

```
# mount /dev/mapper/eui.00001ffe5a94ff8500a0980000af4444 /mnt/ext4
```

Create filesystems (RHEL 8, RHEL 9, and SLES 15)

For RHEL 8, RHEL 9, and SLES 15, you create a filesystem on the native nvme device and mount the filesystem.

Steps

1. Run the `multipath -ll` command to get a list of nvme devices.

```
# multipath -ll
```

The result of this command can be used to find the devices associated `/dev/disk/by-id/nvme-eui.[uuid#]` location. For the example below this would be `/dev/disc/by-id/nvme-eui.000082dd5c05d39300a0980000a52225`.

```
eui.000082dd5c05d39300a0980000a52225 [nvme]:nvme0n6 NVMe,NetApp E-
Series,08520000
size=4194304 features='n/a' hwhandler='ANA' wp=rw
|-+- policy='n/a' prio=50 status=optimized
| `- 0:0:1 nvme0c0n1 0:0 n/a optimized     live
|-+- policy='n/a' prio=50 status=optimized
| `- 0:1:1 nvme0c1n1 0:0 n/a optimized     live
|-+- policy='n/a' prio=10 status=non-optimized
| `- 0:2:1 nvme0c2n1 0:0 n/a non-optimized live
`-+- policy='n/a' prio=10 status=non-optimized
  `- 0:3:1 nvme0c3n1 0:0 n/a non-optimized live
```

2. Create a file system on the partition for the desired nvme device using the location `/dev/disk/by-id/nvme-eui.[id#]`.

   The method for creating a file system varies depending on the file system chosen. This example shows creating an `ext4` file system.

```
# mkfs.ext4 /dev/disk/by-id/nvme-eui.000082dd5c05d39300a0980000a52225
mke2fs 1.42.11 (22-Oct-2019)
Creating filesystem with 2620928 4k blocks and 655360 inodes
Filesystem UUID: 97f987e9-47b8-47f7-b434-bf3ebbe826d0
Superblock backups stored on blocks:
        32768, 98304, 163840, 229376, 294912, 819200, 884736, 1605632

Allocating group tables: done
Writing inode tables: done
Creating journal (32768 blocks): done
Writing superblocks and filesystem accounting information: done
```

3. Create a folder to mount the new device.

```
# mkdir /mnt/ext4
```

4. Mount the device.

```
# mount /dev/disk/by-id/nvme-eui.000082dd5c05d39300a0980000a52225
/mnt/ext4
```

Verify storage access on the host

Before using the namespace, verify that the host can write data to the namespace and

read it back.

What you'll need

An initialized namespace that is formatted with a file system.

Steps

1. On the host, copy one or more files to the mount point of the disk.
2. Copy the files back to a different folder on the original disk.
3. Run the `diff` command to compare the copied files to the originals.

After you finish

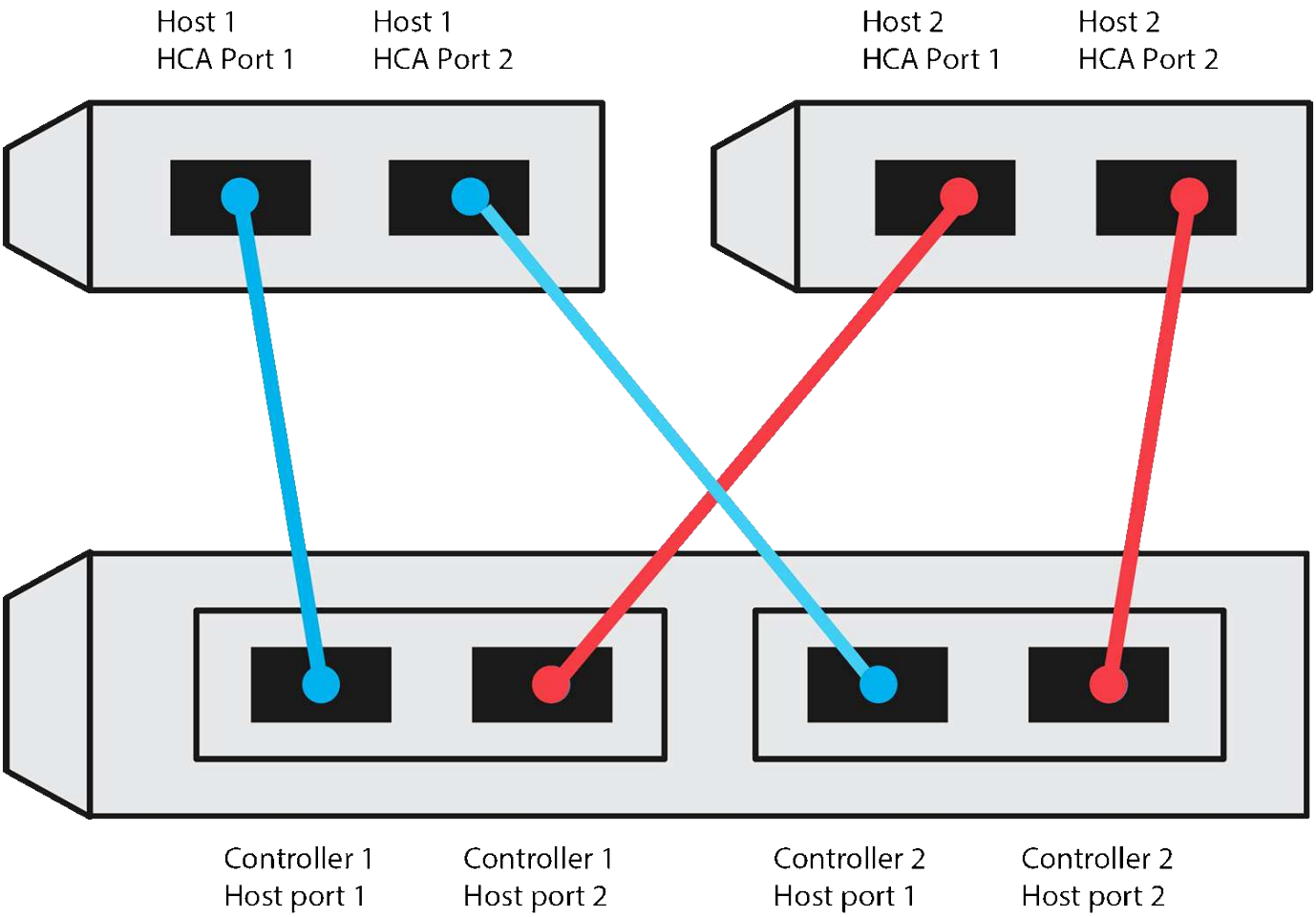
You remove the file and folder that you copied.

Record your NVMe over RoCE configuration

You can generate and print a PDF of this page, and then use the following worksheet to record NVMe over RoCE storage configuration information. You need this information to perform provisioning tasks.

Direct connect topology

In a direct connect topology, one or more hosts are directly connected to the subsystem. In the SANtricity OS 11.50 release, we support a single connection from each host to a subsystem controller, as shown below. In this configuration, one HCA (host channel adapter) port from each host should be on the same subnet as the E-Series controller port it is connected to, but on a different subnet from the other HCA port.

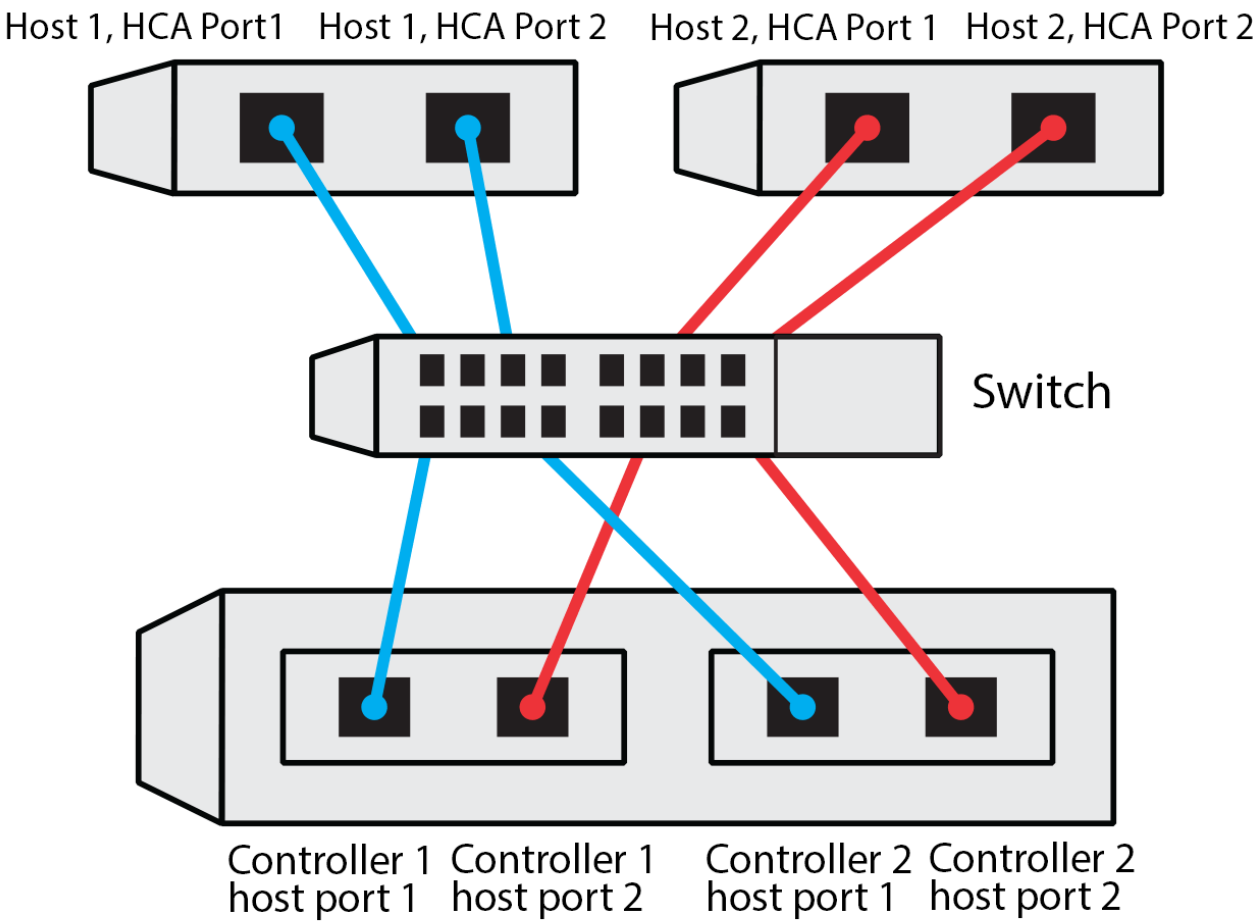


An example configuration that satisfies the requirements consists of four network subnets as follows:

- Subnet 1: Host 1 HCA Port 1 and Controller 1 Host port 1
- Subnet 2: Host 1 HCA Port 2 and Controller 2 Host port 1
- Subnet 3: Host 2 HCA Port 1 and Controller 1 Host port 2
- Subnet 4: Host 2 HCA Port 2 and Controller 2 Host port 2

Switch connect topology

In a fabric topology, one or more switches are used. Refer to NetApp Interoperability Matrix Tool for a list of supported switches.



Host identifiers

Locate and document the initiator NQN from each host.

Host port connections	Software initiator NQN
Host (initiator) 1	
Host (initiator) 2	

Target NQN

Document the target NQN for the storage array.

Array name	Target NQN
Array controller (target)	

Target NQNs

Document the NQNs to be used by the array ports.

Array controller (target) port connections	NQN
Controller A, port 1	
Controller B, port 1	
Controller A, port 2	
Controller B, port 2	

Mapping host name

The mapping host name is created during the workflow.

Mapping host name	
Host OS type	

Copyright information

Trademark information