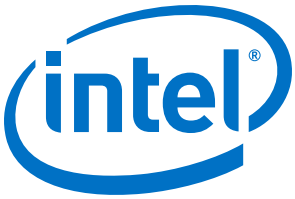


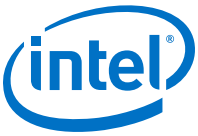
Configuring Non-Volatile Memory Express* (NVMe*) over Fabrics on Intel® Omni-Path Architecture

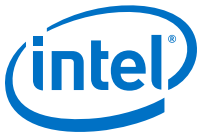
Application Note

Rev. 3.0

January 2020

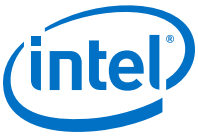
Doc. No.: J78967, Rev.: 3.0





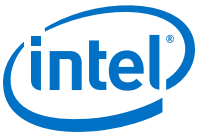
Revision History

Date	Revision	Description
January 2020	3.0	Content rewritten for better focus and usability.
December 2019	2.0	Updates to this document include: • Globally refreshed for newer OSes. • Added Preface. • Updated Introduction and Overview to be consistent with other publications in the document set. • Updated Stop the Target System to include `export NSID=1`. • Added new section, Using SPDK for NVMe over Fabrics Stack.
October 2017	1.0	Initial release of document.



Contents

Figures



Tables

Preface

This manual is part of the documentation set for the Intel® Omni-Path Fabric (Intel® OP Fabric), which is an end-to-end solution consisting of Intel® Omni-Path Host Fabric Interfaces (HFIs), Intel® Omni-Path switches, and fabric management and development tools.

The Intel® OP Fabric delivers the next generation, High-Performance Computing (HPC) network solution that is designed to cost-effectively meet the growth, density, and reliability requirements of large-scale HPC clusters.

Both the Intel® OP Fabric and standard InfiniBand* (IB) are able to send Internet Protocol (IP) traffic over the fabric, or *IPoFabric*. In this document, however, it may also be referred to as *IP over IB* or *IPoIB*. From a software point of view, IPoFabric behaves the same way as IPoIB, and in fact uses an ib_ipoib driver to send IP traffic over the ib0/ib1 ports.

Intended Audience

The intended audience for the Intel® Omni-Path (Intel® OP) document set is network administrators and other qualified personnel.

Intel® Omni-Path Documentation Library

Intel® Omni-Path publications are available at the following URL, under *Latest Release Library*:

https://www.intel.com/content/www/us/en/design/products-and-solutions/networking-and-io/fabric-products/omni-path/downloads.html

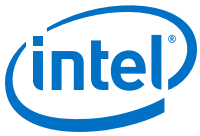
Use the tasks listed in this table to find the corresponding Intel® Omni-Path document.

Task	Document Title	Description
Using the Intel® OPA documentation set	*Intel® Omni-Path Fabric Quick Start Guide*	A roadmap to Intel's comprehensive library of publications describing all aspects of the product family. This document outlines the most basic steps for getting your Intel® Omni-Path Architecture (Intel® OPA) cluster installed and operational.
Setting up an Intel® OPA cluster	*Intel® Omni-Path Fabric Setup Guide*	Provides a high level overview of the steps required to stage a customer-based installation of the Intel® Omni-Path Fabric. Procedures and key reference documents, such as Intel® Omni-Path user guides and installation guides, are provided to clarify the process. Additional commands and best known methods are defined to facilitate the installation process and troubleshooting.

continued...

Task	Document Title	Description
Installing hardware	*Intel® Omni-Path Fabric Switches Hardware Installation Guide*	Describes the hardware installation and initial configuration tasks for the Intel® Omni-Path Switches 100 Series. This includes: Intel® Omni-Path Edge Switches 100 Series, 24 and 48-port configurable Edge switches, and Intel® Omni-Path Director Class Switches 100 Series.
	Intel® Omni-Path Host Fabric Interface Installation Guide	Contains instructions for installing the HFI in an Intel® OPA cluster.
Installing host software Installing HFI firmware Installing switch firmware (externally-managed switches)	*Intel® Omni-Path Fabric Software Installation Guide*	Describes using a Text-based User Interface (TUI) to guide you through the installation process. You have the option of using command line interface (CLI) commands to perform the installation or install using the Linux* distribution software.
Managing a switch using Chassis Viewer GUI Installing switch firmware (managed switches)	*Intel® Omni-Path Fabric Switches GUI User Guide*	Describes the graphical user interface (GUI) of the Intel® Omni-Path Fabric Chassis Viewer GUI. This document provides task-oriented procedures for configuring and managing the Intel® Omni-Path Switch family. Help: GUI embedded help files
Managing a switch using the CLI Installing switch firmware (managed switches)	*Intel® Omni-Path Fabric Switches Command Line Interface Reference Guide*	Describes the command line interface (CLI) task information for the Intel® Omni-Path Switch family. Help: -help for each CLI
Managing a fabric using FastFabric	*Intel® Omni-Path Fabric Suite FastFabric User Guide*	Provides instructions for using the set of fabric management tools designed to simplify and optimize common fabric management tasks. The management tools consist of Text-based User Interface (TUI) menus and command line interface (CLI) commands. Help: -help and man pages for each CLI. Also, all host CLI commands can be accessed as console help in the Fabric Manager GUI.
Managing a fabric using Fabric Manager	*Intel® Omni-Path Fabric Suite Fabric Manager User Guide*	The Fabric Manager uses a well defined management protocol to communicate with management agents in every Intel® Omni-Path Host Fabric Interface (HFI) and switch. Through these interfaces the Fabric Manager is able to discover, configure, and monitor the fabric.
	Intel® Omni-Path Fabric Suite Fabric Manager GUI User Guide	Provides an intuitive, scalable dashboard and set of analysis tools for graphically monitoring fabric status and configuration. This document is a user-friendly alternative to traditional command-line tools for day-to-day monitoring of fabric health. Help: Fabric Manager GUI embedded help files
Configuring and administering Intel® HFI and IPoIB driver Running MPI applications on Intel® OPA	*Intel® Omni-Path Fabric Host Software User Guide*	Describes how to set up and administer the Host Fabric Interface (HFI) after the software has been installed. The audience for this document includes cluster administrators and Message-Passing Interface (MPI) application programmers.
Writing and running middleware that uses Intel® OPA	*Intel® Performance Scaled Messaging 2 (PSM2) Programmer's Guide*	Provides a reference for programmers working with the Intel® PSM2 Application Programming Interface (API). The Performance Scaled Messaging 2 API (PSM2 API) is a low-level user-level communications interface.

continued...



Task	Document Title	Description
Optimizing system performance	*Intel® Omni-Path Fabric Performance Tuning User Guide*	Describes BIOS settings and parameters that have been shown to ensure best performance, or make performance more consistent, on Intel® Omni-Path Architecture. If you are interested in benchmarking the performance of your system, these tips may help you obtain better performance.
Designing an IP or LNet router on Intel® OPA	*Intel® Omni-Path IP and LNet Router Design Guide*	Describes how to install, configure, and administer an IPoIB router solution (Linux* IP or LNet) for inter-operating between Intel® Omni-Path and a legacy InfiniBand* fabric.
Building Containers for Intel® OPA fabrics	*Building Containers for Intel® Omni-Path Fabrics using Docker* and Singularity* Application Note*	Provides basic information for building and running Docker* and Singularity* containers on Linux*-based computer platforms that incorporate Intel® Omni-Path networking technology.
Writing management applications that interface with Intel® OPA	*Intel® Omni-Path Management API Programmer's Guide*	Contains a reference for programmers working with the Intel® Omni-Path Architecture Management (Intel OPAMGT) Application Programming Interface (API). The Intel OPAMGT API is a C-API permitting in-band and out-of-band queries of the FM's Subnet Administrator and Performance Administrator.
Using NVMe* over Fabrics on Intel® OPA	*Configuring Non-Volatile Memory Express* (NVMe*) over Fabrics on Intel® Omni-Path Architecture Application Note*	Describes how to implement a simple Intel® Omni-Path Architecture-based point-to-point configuration with one target and one host server.
Learning about new release features, open issues, and resolved issues for a particular release	*Intel® Omni-Path Fabric Software Release Notes*	
	Intel® Omni-Path Fabric Manager GUI Release Notes	
	Intel® Omni-Path Fabric Switches Release Notes (includes managed and externally-managed switches)	
	Intel® Omni-Path Fabric Unified Extensible Firmware Interface (UEFI) Release Notes	
	Intel® Omni-Path Fabric Thermal Management Microchip (TMM) Release Notes	
	Intel® Omni-Path Fabric Firmware Tools Release Notes	

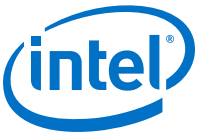
How to Search the Intel® Omni-Path Documentation Set

Many PDF readers, such as Adobe* Reader and Foxit* Reader, allow you to search across multiple PDFs in a folder.

Follow these steps:

1. Download and unzip all the Intel® Omni-Path PDFs into a single folder.
2. Open your PDF reader and use **CTRL-SHIFT-F** to open the Advanced Search window.
3. Select **All PDF documents in...**
4. Select **Browse for Location** in the dropdown menu and navigate to the folder containing the PDFs.
5. Enter the string you are looking for and click **Search**.

Use advanced features to further refine your search criteria. Refer to your PDF reader Help for details.



Cluster Configurator for Intel® Omni-Path Fabric

The Cluster Configurator for Intel® Omni-Path Fabric is available at: http://www.intel.com/content/www/us/en/high-performance-computing-fabrics/omni-path-configurator.html.

This tool generates sample cluster configurations based on key cluster attributes, including a side-by-side comparison of up to four cluster configurations. The tool also generates parts lists and cluster diagrams.

Documentation Conventions

The following conventions are standard for Intel® Omni-Path documentation:

- *Note:* provides additional information.
- ***Caution:*** indicates the presence of a hazard that has the potential of causing damage to data or equipment.
- ***Warning:*** indicates the presence of a hazard that has the potential of causing personal injury.
- Text in blue font indicates a hyperlink (jump) to a figure, table, or section in this guide. Links to websites are also shown in blue. For example:

  See License Agreements on page 11 for more information.

  For more information, visit www.intel.com.
- Text in **bold** font indicates user interface elements such as menu items, buttons, check boxes, key names, key strokes, or column headings. For example:

  Click the **Start** button, point to **Programs**, point to **Accessories**, and then click **Command Prompt**.

  Press **CTRL+P** and then press the **UP ARROW** key.
- Text in `Courier` font indicates a file name, directory path, or command line text. For example:

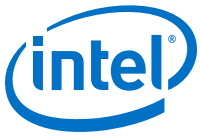
  Enter the following command: `sh ./install.bin`
- Text in *italics* indicates terms, emphasis, variables, or document titles. For example:

  Refer to *Intel® Omni-Path Fabric Software Installation Guide* for details.

  In this document, the term *chassis* refers to a managed switch.

Procedures and information may be marked with one of the following qualifications:

- **(Linux)** – Tasks are only applicable when Linux* is being used.
- **(Host)** – Tasks are only applicable when Intel® Omni-Path Fabric Host Software or Intel® Omni-Path Fabric Suite is being used on the hosts.
- **(Switch)** – Tasks are applicable only when Intel® Omni-Path Switches or Chassis are being used.
- Tasks that are generally applicable to all environments are not marked.



Best Practices

- Intel recommends that users update to the latest versions of Intel® Omni-Path firmware and software to obtain the most recent functional and security updates.
- To improve security, the administrator should log out users and disable multi-user logins prior to performing provisioning and similar tasks.

License Agreements

This software is provided under one or more license agreements. Please refer to the license agreement(s) provided with the software for specific detail. Do not install or use the software until you have carefully read and agree to the terms and conditions of the license agreement(s). By loading or using the software, you agree to the terms of the license agreement(s). If you do not wish to so agree, do not install or use the software.

Technical Support

Technical support for Intel® Omni-Path products is available 24 hours a day, 365 days a year. Please contact Intel Customer Support or visit http://www.intel.com/omnipath/support for additional detail.

1.0 Introduction

This application note describes how to implement Non-Volatile Memory Express* (NVMe*) over Fabrics on the Intel® Omni-Path Architecture. Focusing on a simple implementation, it describes how to install and set up a point-to-point configuration with one target and one host server.

NOTE

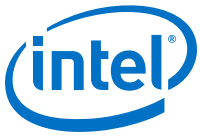
The concepts in this document can be expanded to more complex topologies.

1.1 Terminology

The following table lists the unique terms and acronyms found in this document.

Table 1. Terminology

Term	Description
HPC	High Performance Computing
IPoIB	IP over InfiniBand*
LUN	Logical Unit Number
NVMe*	Non-Volatile Memory Express*
RDMA	Remote Direct Memory Access
RoCE	RDMA over Converged Ethernet
SRP	Secure Remote Password protocol
SSD	Solid State Drive
iSCSI	Internet Small Computer Systems Interface



2.0 Overview

The NVM Express specification includes support for NVMe over Fabrics such as Ethernet*, InfiniBand* (IB), and Intel® Omni-Path. The NVMe over Fabrics specification does not rely on a particular hardware interface. It works with most RDMA-enabled fabrics such as Ethernet (iWarp*, RoCE*), InfiniBand, or Intel® Omni-Path fabric.

NVMe over Fabrics allows for a network-based storage system. This permits NVMe storage devices to be remote from the server that is using them. In many regards, NVMe over Fabrics is similar to previous storage protocols such as iSCSI, SRP, and Fibre Channel. However, unlike these legacy protocols, NVMe over Fabrics has been optimized to take advantage of the low latency and high performance characteristics of modern NVMe solid state storage devices (SSD). To enable this, a high-speed interface between client and server is required, and a special protocol is used between them. Using NVMe over Fabrics, you can attach and detach NVMe SSDs to the clients (host initiators) based on the requirements of a given workload or application.

You can also partition the SSD and slice it into pieces or aggregate multiple SSDs if the workload requires it. Alternatively, you can design multi-path configurations where a storage device (target) is attached to multiple host initiators at the same time. This configuration flexibility has advantages for many uses in High Performance Computing (HPC) architecture design.

For an overview of NVMe over Fabrics, refer to http://www.nvmexpress.org/wp-content/uploads/NVMe_Over_Fabrics.pdf.

Additional information may be found at:

- NVM Express Home Page:

  http://nvmexpress.org/

- NVM Express over Fabrics specification:

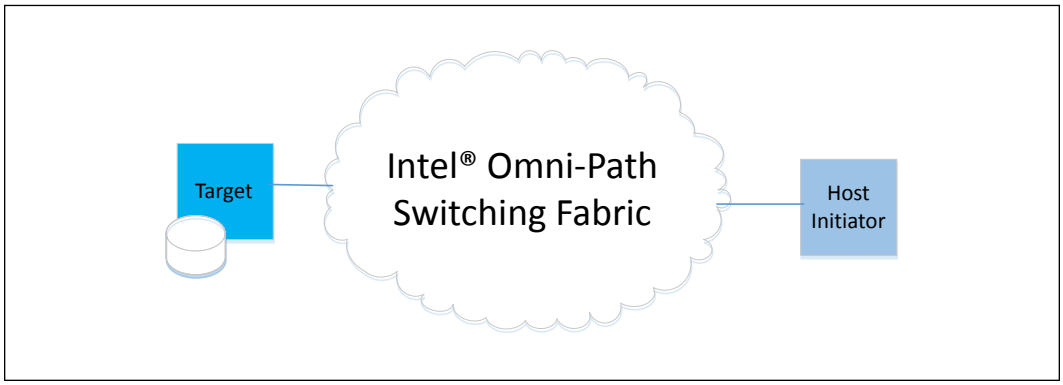
  https://nvmexpress.org/wp-content/uploads/NVMe-over-Fabrics-1.1-2019.10.22-Ratified.pdf

Similar to the legacy protocols, NVMe over Fabrics has two basic roles:

- A *target* (or controller side) is a storage device that offers fabric access to some or all of its NVMe storage.
- A *host initiator* (or client side) is a storage client that reads and/or writes to the remote storage.

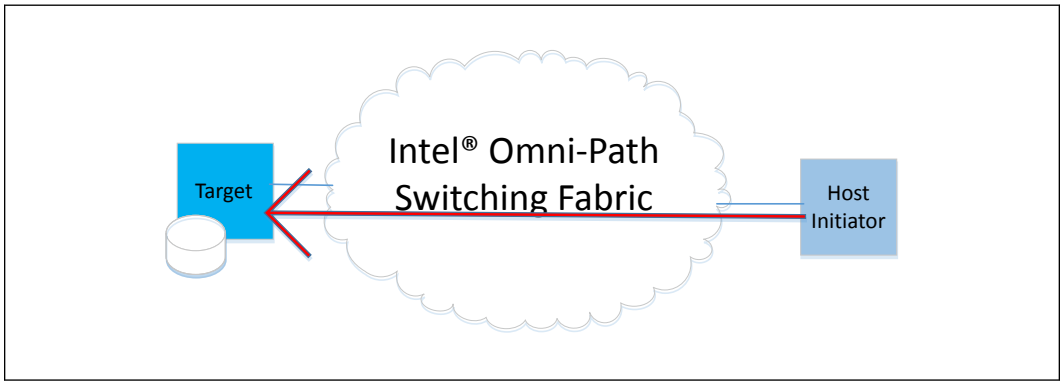
For Intel® Omni-Path fabrics, the host initiator and the target are typically servers, both of which must contain an HFI. The following figure shows a simple configuration with a single host initiator and a single target.

Figure 1. Simple Configuration



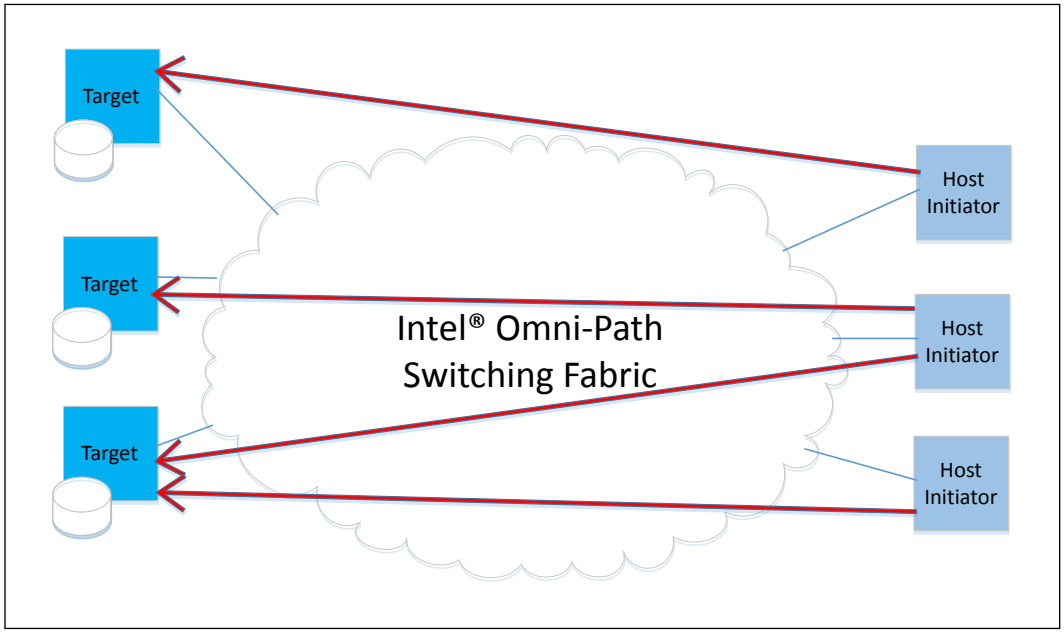
Target devices are passive. The association between host initiator and target is established by the host initiator and accepted by the target as shown in the following figure.

Figure 2. Basic Initiator and Target Association



Using these basic concepts, complex configurations can be created with many associations between host initiators and targets. As is shown in the following figure, a host initiator may connect to multiple targets. Similarly, a target may accept connections from multiple host initiators. If a target is accessed by multiple host initiators, typically each host initiator will access a different SSD or SSD partition within the target.

Figure 3. Complex Configuration



3.0 Using NVMe over Fabrics with Intel® OPA

NVMe over Fabrics is supported on the following Linux distributions:

- RHEL* 7.4 and newer
- CentOS*-7 (1708) and newer
- SLES* 12 SP3 and newer

When using NVMe over Fabrics with Intel® Omni-Path, it is recommended to use Intel® Omni-Path Fabric Software Release 10.9.3.1.1 or later on one of the distributions listed above.

3.1 Linux Distribution Installation

When installing the Linux distribution, be sure to install the following packages.

- On the target, install:
  - `nvmetcli`

    This package includes the `nvmetcli` command that can simplify configuration of NVMe targets.

    If this package is not available, you can choose to manually configure the target through instructions provided in this document, or you can download `nvmetcli` from http://git.infradead.org/users/hch/nvmetcli.git.
- On the initiator, install:
  - `nvme-cli`

    This package contains tools that simplify the target discovery and connection process.

    If this package is not available for your distribution, you can download and install it from https://github.com/linux-nvme/nvme-cli.

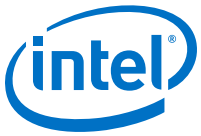
3.2 Intel® Omni-Path Installation

Intel® Omni-Path should be installed and verified following the normal installation procedures as documented in *Intel® Omni-Path Fabric Setup Guide*.

NVMe over Fabrics will require that IP over InfiniBand* (IPoIB) is configured for the Intel® Omni-Path HFI. Follow the instructions in the *Intel® Omni-Path Fabric Performance Tuning User Guide* to configure Intel® Omni-Path for optimal verbs performance, especially on NVMe over Fabrics targets.

If possible, Intel recommends the following to optimize performance: On the target, place the Intel® Omni-Path HFI and the NVMe SSDs on the same CPU socket.

In addition to the fabric verification procedure from the *Intel® Omni-Path Fabric Setup Guide*, the following simple test may be performed to check HFI verbs connectivity and performance.



On the host initiator:

```
# ib_write_bw -F -R -s 1048576
```

On the target:

```
# ib_write_bw -F -R -s 1048576 initiator_ipoib_address
```

This test performs 1 MB RDMA writes from the target to the host initiator, simulating the fabric traffic for storage reads. The performance should be close to or exceed 12 GB/s at 1 MB message sizes using Intel® Xeon® Processors with Intel® Turbo Boost Technology enabled.

4.0 Setting Up NVMe over Fabrics

Setting up NVMe over Fabrics is the same as for most RDMA fabrics. When using Intel® Omni-Path, ensure the following settings:

- Use the Intel® Omni-Path IPoIB IP address for the NVMe over Fabrics target IP address.
- Set the NVMe over Fabrics transport type to `rdma`.

Documentation for setting up NVMe may be found from your Linux distribution provider, such as: https://documentation.suse.com/sles/15-SP1/html/SLES-all/cha-nvmeof.html#sec-nvmeof-target-configuration.

The steps provided in the following sections set up a simple, example configuration with a single initiator and a single target.

4.1 Set Up the NVMe over Fabrics Target

This example uses the `/dev/nvme0n1` NVMe SSD for NVMe over Fabrics. It assumes the NVMe device has already been installed and formatted. This example configures an NVMe subsystem with a single logical unit number (LUN). For simplicity in this example, the target is configured so any host initiator may access it. The basic configuration parameters used in this example are:

Table 2. NVMe over Fabrics Target Parameters

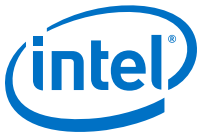
Parameter	Value	Comments
NVMe over Fabrics Port Number (`trsvcid`)	4420	This is the default.
NVMe Target Server's Intel® OPA IPoIB IP Address	10.10.10.20	Use the proper Intel® Omni-Path IPoIB IP address for your target node.
NVMe over Fabrics Qualified Name (NQN)	mydrive	This name is used for simplicity. Choose a more meaningful name and naming convention for production uses.

Using the nvmetcli Tool

The easiest way to set up NVMe over Fabrics is with the `nvmetcli` tool. The man page provides more details. Once an NVMe over Fabrics target has been set up, the configuration file may be saved and restored later, such as after a target server reboot.

1. If not already loaded, load the `nvme` and `nvmet` (target) drivers:

```
# modprobe configfs
# modprobe nvme
# modprobe nvmet
# modprobe nvmet_rdma
```



NOTE

On some Linux distributions, the `configfs` kernel module may not be available, in which case it need not be loaded. On most distributions, loading `nvmet_rdma` will automatically load `nvmet`.

2. Launch the `nvmetcli` tool as root:

```
# nvmetcli
```

3. Create an NVMe over Fabrics port, subsystem, and namespace with the NVMe device:

```
/> cd subsystems
/subsystems> create mydrive
/subsystems> cd mydrive/namespaces
/subsystems/mydrive/namespaces> create 1
/subsystems/mydrive/namespaces> cd 1
/subsystems/mydrive/namespaces/1> set device path=/dev/nvme0n1
Parameter path is now '/dev/nvme0n1'.
/subsystems/mydrive/namespaces/1> cd ../..
```

For simplicity, in this example we will allow any host to use this target:

```
/subsystems/mydrive> set attr allow_any_host=1
Parameter allow_any_host is now '1'.
/subsystems/mydrive> cd namespaces/1
/subsystems/mydrive/namespaces/1> enable
The Namespace has been enabled.
/subsystems/mydrive/namespaces/1> cd ../../../..
/> cd ports
/ports> create 1
/ports> cd 1
/ports/1> set addr trtype=rdma adrfam=ipv4 trsvcid=4420 traddr=10.10.10.20
Parameter trtype is now ‘rdma’.
Parameter adrfam is now ‘ipv4’.
Parameter trsvcid is now ‘4420’.
Parameter traddr is now ’10.10.10.20’.
/ports/1> cd subsystems
/ports/1/subsystems> create mydrive
```

NOTE

If the `create mydrive` command fails with the error `Could not symlink mydrive in configFS: [Errno 99] Cannot assign requested address`, this may indicate that the IP address supplied for `traddr` is incorrect or that IPoIB is not *up* on the target node.

Confirm IPoIB is up using ping and verify that the correct IP address is entered.

4. Save the target configuration to a file:

```
/ports/1/subsystems> saveconfig myconfig.json
/ports/1/subsystems> exit
```

As needed, to restore the saved configuration, use the following command:

```
# nvmetcli restore myconfig.json
```

NOTE

If desired, the nvme target configuration can be enabled for automatic restoration on server reboot. Refer to the `nvmetcli` man page for details.

Using Command Line (Manual Method)

The following example sets up the same NVMe over Fabrics target as above, however it does not use the `nvmetcli` tool. Instead, it directly creates the necessary files and settings in `/sys/kernel/config/nvmet/`. In this example, environment variables are used to hold the configurable parameters, so you may use this same sequence of commands with different values for the environment variables as needed.

```
# export TARGET=10.10.10.20
# export DEV=/dev/nvme0n1
# export PORT=4420
# export NVME_PORT=1
# export NQN=mydrive
# export NSID=1
# modprobe configfs
# modprobe nvme
# modprobe nvmet
# modprobe nvmet_rdma
# cd /sys/kernel/config/nvmet/subsystems/
# echo creating ${NQN} on device ${DEV}
# mkdir ${NQN}
# sleep 1
# mkdir ${NQN}/namespaces/${NSID}
# echo -n ${DEV} > ${NQN}/namespaces/${NSID}/device_path
# echo -n 1 > ${NQN}/attr_allow_any_host
# echo -n 1 > ${NQN}/namespaces/${NSID}/enable
# cd /sys/kernel/config/nvmet/ports
# mkdir ${NVME_PORT}
# echo -n ipv4 > ${NVME_PORT}/addr_adrfam
# echo -n rdma > ${NVME_PORT}/addr_trtype
# echo -n not required > ${NVME_PORT}/addr_treq
# echo -n ${TARGET} > ${NVME_PORT}/addr_traddr
# echo -n ${PORT} > ${NVME_PORT}/addr_trsvcid
# ln -s ../subsystems/${NQN} ${NVME_PORT}/subsystems/${NQN}
```

4.2 Connect the Host Initiator to the Target

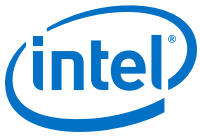
Once the target device is set up, you can have a host initiator discover and connect to the target. This example assumes the target was set up using one of the two approaches and configuration parameters shown in the previous section.

1. If not already loaded, load the NVMe over Fabrics drivers:

```
# modprobe nvme_rdma
# modprobe nvme_fabrics
# modprobe nvme_core
```

NOTE

On most Linux distributions, loading `nvme_rdma` will automatically load `nvme_fabrics` and `nvme_core`.



2. Use the `nvme` command to discover the target device using the IP address and port number of the target.

   Use the results to confirm that the target is properly set up and ready for use.

```
# nvme discover -t rdma -s 4420 -a 10.10.10.20
Discovery Log Number of Records 1, Generation counter 1
=====Discovery Log Entry 0======
trtype: rdma
adrfam: ipv4
subtype: nvme subsystem
treq: not required
portid: 1
trsvcid: 4420
subnqn: mydrive
traddr: 10.10.10.20
rdma_prtype: unrecognized
rdma_qptype: connected
rdma_cms: rdma-cm
rdma_pkey: 0x0000
```

3. Establish a connection to the target using the IP address, NQN and port number.

```
# nvme connect -t rdma -n mydrive -s 4420 -i 32 -a 10.10.10.20
```

   In the above example, we also chose the use of 32 queue pairs through the `-i` option. Using multiple queue pairs can provide improved performance and functionality for NVMe over Fabrics. For most configurations, 32 queue pairs will work best.

   NOTE

   You can review other available connection options using `nvme connect --help`.

   Once the connection is established, the NVMe over Fabrics device will appear as if it is a local NVMe device (`/dev/nvme0 and /dev/nvme0n1`).

```
# ls /dev/nvm*
/dev/nvme0 /dev/nvme0n1 /dev/nvme-fabrics
```

4. At this point, you may create file systems and mount the device the same as you would for any local NVMe device.

   Many choices are available for accomplishing these tasks. The following example shows how to create an XFS configuration with 4k sector size and disabled DISCARD flag:

```
mkfs -t xfs -K -s size=4096 -m crc=0 /dev/nvme0n1
```

   Then, to mount it for use:

```
mount /dev/nvme0n1 /mnt/data -o nobarrier,noatime,nodiratime
```

   NOTE

   If `nvme0n1` has been partitioned, you may want to use `/dev/nvme0n1p1` for the `mkfs` and `mount` commands on the initiator.

4.3 Disconnect the Initiator

When the host initiator is done using the device, the NVMe over Fabrics connection can be disconnected.

1. Stop any applications and unmount any file systems on the device prior to disconnecting it.
2. To disconnect the host initiator:

```
# umount /dev/nvme0n1
# nvme disconnect –n mydrive
```

4.4 Stop the Target System

Use one of the methods below to stop the target system.

Using the nvmetcli Tool

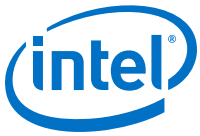
The target devices can be easily de-configured using:

```
# nvmetcli clear
```

Using Command Line (Manual Method)

You can de-configure the target devices by performing the following:

```
# export NVME_PORT=1
# export NQN=mydrive
# export NSID=1
# cd /sys/kernel/config/nvmet/ports
# rm ${NVME_PORT}/subsystems/*
# rmdir ${NVME_PORT}
# cd /sys/kernel/config/nvmet/subsystems/
# echo -n 0 > ${NQN}/namespaces/${NSID}/enable
# rmdir ${NQN}/namespaces/${NSID}
# rmdir ${NQN}
If desired, the nvme over fabrics drivers may be unloaded:
# modprobe -r nvmet_rdma
# modprobe -r nvmet
```



5.0 Using SPDK for NVMe over Fabrics Stack

The Storage Performance Development Kit (SPDK), a collection of application software acceleration tools and libraries, was developed by Intel to accelerate the use of NVMe SSDs as a back-end storage solution. The core of this software library is a user space, asynchronous, poll mode NVMe driver. Compared to kernel NVMe drivers, this driver provides a cost-effective solution that can greatly reduce the latency of NVMe commands and improve input/output operations per second per-CPU core.

For more information on using SPDK for NVMe over Fabrics, refer to the following:

- Storage Performance Development Kit:

  https://spdk.io/

- Accelerate Your NVMe Drives with SPDK:

  https://software.intel.com/en-us/articles/accelerating-your-nvme-drives-with-spdk

- Introduction to the Storage Performance Development Kit (SPDK):

  https://software.intel.com/en-us/articles/introduction-to-the-storage-performance-development-kit-spdk