



RS4243AS Lossless Test Direct Attach Global Pause

V1.00 - August 27, 2026

Copyright 2026 HighPoint Technologies, Inc.

All rights reserved

Table of Contents

1. Overview	2
2. Test Environment	3
3. Lossless Network Setup Steps	4
3.1. NVMe-oF Controller (C2000) Configuration	4
3.2. Mellanox ConnectX-6 Dx Configuration	5
3.3. Ubuntu 24.04 OS Configuration	7
3.3.1. Enable Global Pause	7
3.3.2. Limit CPU states to 0 and 1	8
3.3.3. Switch to latency-performance mode	9
3.3.4. Set up network card port information	9
4. FIO Test	10
4.1. Connect the SSDs	10
4.2. Format the SSDs	12
4.3. FIO Test Scripts	13
4.3.1. Single-port test scripts	13
4.3.2. Dual-port test scripts	14
4.4. Performance Summary	16
4.4.1. Single-port test results	16
4.4.2. Dual-port test results	16

1. Overview

This report presents the lossless test results for the RS4243AS platform under Direct Attach and Global Pause configurations. The test validates the platform's ability to maintain stable, zero-packet-loss data transmission in high-throughput NVMe-oF storage environments. Performance is evaluated using FIO workloads across single-port and dual-port setups with 24 NVMe SSDs. Results confirm that RS4243AS, with Global Pause properly configured, delivers consistent sequential read and write performance with minimal fluctuation, making it suitable for latency-sensitive and mission-critical storage applications.

2. Test Environment

Motherboard	Dell PowerEdge R660																																																																														
CPU	Intel Xeon Silver 4514Y*2																																																																														
Memory	256GB (32GB*8)																																																																														
BIOS	2.8.2																																																																														
SLOT	SLOT1																																																																														
Product	Chassis: RS4243AS																																																																														
Mellanox	ConnectX-6 Dx-MCX623106AN-CDAT (Firmware version: 22.43.4100)																																																																														
Cable	Mellanox-MCP1600-C003 *2																																																																														
SSD	SANDISK SN655 30.72TB *24																																																																														
SSD Performance	<table><tr><th colspan="6">Product Information</th></tr><tr><td>Capacity¹</td><td>3.84TB</td><td>7.68TB</td><td>15.36TB</td><td>30.72TB</td><td>61.44TB</td></tr><tr><td>Endurance²</td><td colspan="5">1DWPD</td></tr><tr><td>Security</td><td colspan="5">SE, ISE, TCG</td></tr><tr><td>Form Factor</td><td colspan="5">U.3 15mm</td></tr><tr><td>Interface</td><td colspan="5">PCIe Gen4, NVMe 1.4</td></tr><tr><th colspan="6">Performance³</th></tr><tr><td>Read Throughput (max MB/s, Seq 128KiB)</td><td>6,800MB/s</td><td>6,800MB/s</td><td>6,800MB/s</td><td>6,100MB/s</td><td>4,300MB/s</td></tr><tr><td>Write Throughput (max MB/s, Seq 128KiB)</td><td>2,600MB/s</td><td>2,000MB/s</td><td>3,700MB/s</td><td>3,400MB/s</td><td>3,150MB/s</td></tr><tr><td>Read IOPS (max, Rnd 4KiB)</td><td>1000K</td><td>980K</td><td>1100K</td><td>1052K</td><td>890K</td></tr><tr><td>Write IOPS (max, Rnd 4KiB)</td><td>112K</td><td>80K</td><td>135K</td><td>66K</td><td>29K</td></tr><tr><td>Read Latency (μs)⁴</td><td>80</td><td>78</td><td>125</td><td>110</td><td>115</td></tr><tr><td>Write Latency (μs)⁵</td><td>10</td><td>15</td><td>10</td><td>15</td><td>40</td></tr></table>	Product Information						Capacity ¹	3.84TB	7.68TB	15.36TB	30.72TB	61.44TB	Endurance ²	1DWPD					Security	SE, ISE, TCG					Form Factor	U.3 15mm					Interface	PCIe Gen4, NVMe 1.4					Performance ³						Read Throughput (max MB/s, Seq 128KiB)	6,800MB/s	6,800MB/s	6,800MB/s	6,100MB/s	4,300MB/s	Write Throughput (max MB/s, Seq 128KiB)	2,600MB/s	2,000MB/s	3,700MB/s	3,400MB/s	3,150MB/s	Read IOPS (max, Rnd 4KiB)	1000K	980K	1100K	1052K	890K	Write IOPS (max, Rnd 4KiB)	112K	80K	135K	66K	29K	Read Latency (μs) ⁴	80	78	125	110	115	Write Latency (μs) ⁵	10	15	10	15	40
	Product Information																																																																														
	Capacity ¹	3.84TB	7.68TB	15.36TB	30.72TB	61.44TB																																																																									
	Endurance ²	1DWPD																																																																													
	Security	SE, ISE, TCG																																																																													
	Form Factor	U.3 15mm																																																																													
	Interface	PCIe Gen4, NVMe 1.4																																																																													
	Performance ³																																																																														
	Read Throughput (max MB/s, Seq 128KiB)	6,800MB/s	6,800MB/s	6,800MB/s	6,100MB/s	4,300MB/s																																																																									
	Write Throughput (max MB/s, Seq 128KiB)	2,600MB/s	2,000MB/s	3,700MB/s	3,400MB/s	3,150MB/s																																																																									
	Read IOPS (max, Rnd 4KiB)	1000K	980K	1100K	1052K	890K																																																																									
	Write IOPS (max, Rnd 4KiB)	112K	80K	135K	66K	29K																																																																									
	Read Latency (μs) ⁴	80	78	125	110	115																																																																									
	Write Latency (μs) ⁵	10	15	10	15	40																																																																									
		Data Sheet: Ultrastar DC SN655 NVMe SSD																																																																													
OS	Ubuntu 24.04 server (kernel:6.8.0-134)																																																																														
Mellanox Driver	MLNX_OFED_LINUX-24.10-3.2.5.0-ubuntu24.04-x86_64.tgz																																																																														
Connection method	A Mellanox network card is connected to the RS4243AS chassis using two optical fibers.																																																																														

3. Lossless Network Setup Steps

3.1. NVMe-oF Controller (C2000) Configuration

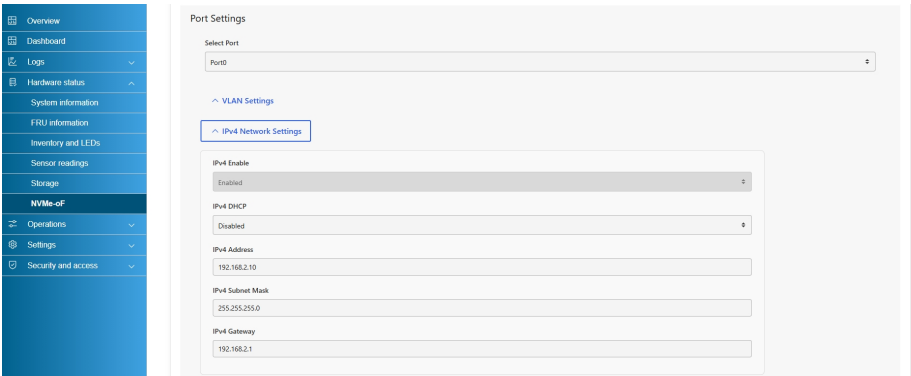
1. The NVMe-oF Controller's default CCMODE is 0; no need to configure it again.

```
CCMode (Congestion Control Mode) MI:  
+ MODE: 0
```

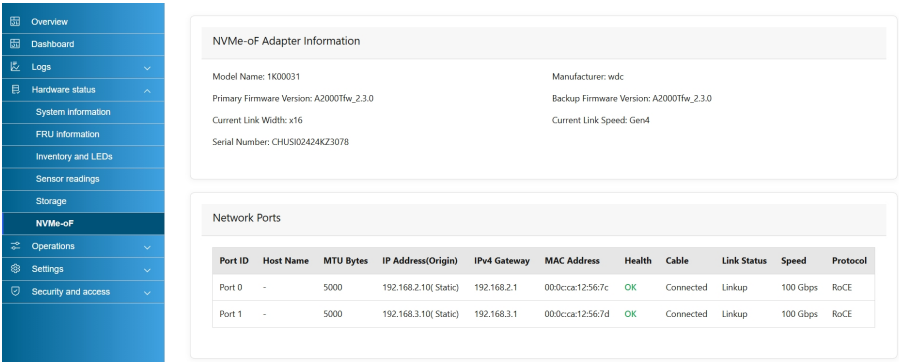
2. Set up C2000 port IP.

Open the HighPoint BMC interface, navigate to Hardware Status → NVMe-oF → Port Settings, and configure the following settings:

- 1) Select port: Select the port you want to configure for the C2000.
- 2) IPv4 Network Settings: Configure the IP settings for the C2000 port, as shown in the figure below:



After configuring the IP settings for the two C2000 ports, the result is shown in the figure below.



3.2. Mellanox ConnectX-6 Dx Configuration

1. Download the official website .tgz driver file.

Link: https://network.nvidia.com/products/infiniband-drivers/linux/mlnx_ofed/

2. Copy the driver file to the host.

```
#cp MNX_OFED_LINUX-x-x.x.x-x86_64.tgz /home/test
```

3. Unzip the driver files on the host.

```
#tar zxvf MNX_OFED_LINUX-x-x.x.x-x86_64.tgz
```

4. Install the driver from the directory where the driver files were extracted.

```
#!/mlnxofedinstall --add-kernel-support --with-nvmf --skip-repo
```

```
root@test:/home/test/MNX_OFED_LINUX-24.10-3.2.5.0-ubuntu24.04-x86_64# ./mlnxofedinstall --add-kernel-support --with-nvmf --skip-repo
Note: This program will create MNX_OFED_LINUX_TGZ for ubuntu24.04 under /tmp/MNX_OFED_LINUX-24.10-3.2.5.0-6.8.0-134-generic directory.
See log file /tmp/MNX_OFED_LINUX-24.10-3.2.5.0-6.8.0-134-generic/mlnx_iso.252697_logs/mlnx_iso.252697.log

Checking if all needed packages are installed...
Building MNX_OFED_LINUX DEBS. Please wait...
Created /tmp/MNX_OFED_LINUX-24.10-3.2.5.0-6.8.0-134-generic/MNX_OFED_LINUX-24.10-3.2.5.0-ubuntu24.04-ext.tgz
Removing old packages...
Installing /tmp/MNX_OFED_LINUX-24.10-3.2.5.0-6.8.0-134-generic/MNX_OFED_LINUX-24.10-3.2.5.0-ubuntu24.04-ext
/tmp/MNX_OFED_LINUX-24.10-3.2.5.0-6.8.0-134-generic/MNX_OFED_LINUX-24.10-3.2.5.0-ubuntu24.04-ext/mlnxofedinstall --force --without-dkms --with-nvmf
Logs dir: /tmp/MNX_OFED_LINUX-488697/logs
General log file: /tmp/MNX_OFED_LINUX-488697_logs/general.log

Below is the list of MNX_OFED_LINUX packages that you have chosen
(some may have been added by the installer due to package dependencies):
ofed-scripts
mlnx-tools
mlnx-ofed-kernel-utils
mlnx-ofed-kernel-modules
user-modules
isert-modules
srp-modules
mlnx-nvme-modules
rdma-core
libibverbs1
ibverbs-utils
ibverbs-providers
libibverbs-dev
libibverbs1-debug
libibmad3
libibmad-dev
```

5. Create or edit the file /etc/modules-load.d/modules.conf to include the following:

```
#Mellanox nvmeOF
```

```
rdma_cm
```

```
ib_uverbs
```

```
rdma_ucm
```

```
nvme
```

```
nvme-rdma
```

```
ib_core
```

```
ib_cm
```

```
ib_umad
```

```
nvme_rdma
```

```
nvme_fabrics
```

```
GNU nano 7.2 /etc/modules-load.d/modules.conf *
# /etc/modules is obsolete and has been replaced by /etc/modules-load.d/.
# Please see modules-load.d(5) and modprobe.d(5) for details.
#
# Updating this file still works, but it is undocumented and unsupported.
kmem

# Mellanox nvmeOF
rdma_cm
ib_uverbs
rdma_ucm
nvme
nvme-rdma
ib_core
ib_cm
ib_umad
nvme_rdma
nvme_fabrics
```

6. Restart the driver.

#/etc/init.d/openibd restart

```
root@test:/home/test/MLNX_OFED_LINUX-24.10-3.2.5.0-ubuntu24.04-x86_64# /etc/init.d/openibd restart
Unloading HCA driver: [ OK ]
Loading HCA driver and Access Layer: [ OK ]
```

7. Restart the host system.

#reboot8. Check `ibv_devices` and make sure the Mellanox network adapter is in the list;**#ibv_devices**

```
root@test:/home/test/MLNX_OFED_LINUX-24.10-3.2.5.0-ubuntu24.04-x86_64# ibv_devices
device          node GUID
-----
mlx5_0          e89e4903007c19c8
mlx5_1          e89e4903007c19c9
root@test:/home/test/MLNX_OFED_LINUX-24.10-3.2.5.0-ubuntu24.04-x86_64#
```

3.3. Ubuntu 24.04 OS Configuration

3.3.1. Enable Global Pause

1. Check current pause frame status of network ports.

```
#ethtool -a enp13s0f0np0
```

```
#ethtool -a enp13s0f1np1
```

```
root@test:/home/test# ethtool -a enp13s0f0np0
Pause parameters for enp13s0f0np0:
Autonegotiate:  off
RX:             off
TX:             off

root@test:/home/test# ethtool -a enp13s0f1np1
Pause parameters for enp13s0f1np1:
Autonegotiate:  off
RX:             off
TX:             off
```

Note: `enp13s0f0np0` and `enp13s0f1np1` are two physical ports of the same Mellanox NIC.

2. Turn on RX and TX.

```
#ethtool -A enp13s0f0np0 rx on tx on
```

```
#ethtool -A enp13s0f1np1 rx on tx on
```

```
root@test:/home/test# ethtool -A enp13s0f0np0 rx on tx on
root@test:/home/test# ethtool -A enp13s0f1np1 rx on tx on
```

3. Verify Global Pause configuration takes effect.

```
#ethtool -a enp13s0f0np0
```

```
#ethtool -a enp13s0f1np1
```

```
root@test:/home/test# ethtool -a enp13s0f0np0
Pause parameters for enp13s0f0np0:
Autonegotiate:  off
RX:             on
TX:             on

root@test:/home/test# ethtool -a enp13s0f1np1
Pause parameters for enp13s0f1np1:
Autonegotiate:  off
RX:             on
TX:             on
```

3.3.2. Limit CPU states to 0 and 1

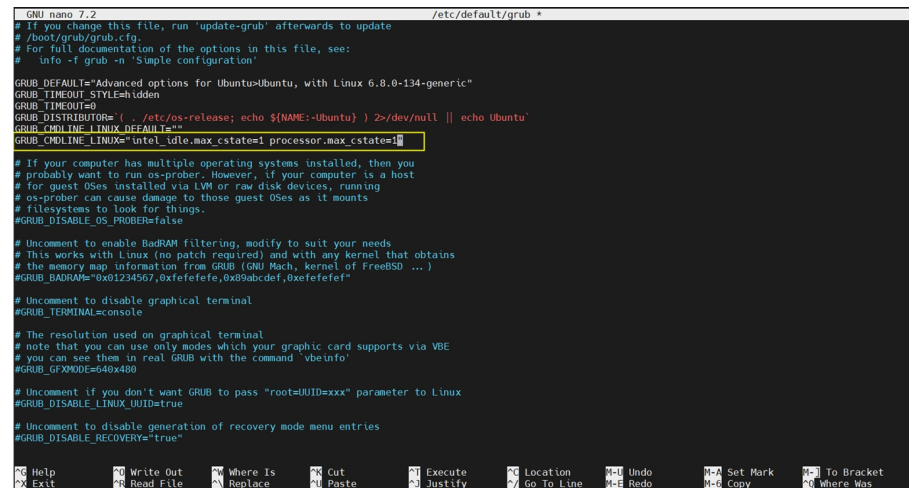
1. Check the default state of CPU states.

```
#cat /sys/devices/system/cpu/cpu0/cpuidle/state*/name
```

```
#cat /sys/devices/system/cpu/cpu0/cpuidle/state*/disable
```

```
root@test:/home/test/0F_Deploy# cat /sys/devices/system/cpu/cpu0/cpuidle/state*/name
POLL
C1
C1E
C6
root@test:/home/test/0F_Deploy# cat /sys/devices/system/cpu/cpu0/cpuidle/state*/disable
0
0
0
0
```

2. Edit the file /etc/default/grub and add the parameters "intel_idle.max_cstate=1 processor.max_cstate=1".



```
GNU nano 7.2 /etc/default/grub
# If you change this file, run 'update-grub' afterwards to update
# /boot/grub/grub.cfg.
# For full documentation of the options in this file, see:
# info -f grub -n 'Simple configuration'

GRUB_DEFAULT="Advanced options for Ubuntu-Ubuntu, with Linux 6.8.0-134-generic"
GRUB_TIMEOUT_STYLE=hidden
GRUB_TIMEOUT=0
GRUB_DISTRIBUTOR="(../etc/os-release; echo ${NAME:-Ubuntu}) 2>/dev/null || echo Ubuntu"
GRUB_CMDLINE_LINUX="intel_idle.max_cstate=1 processor.max_cstate=1"
# If your computer has multiple operating systems installed, then you
# probably want to run os-prober. However, if your computer is a host
# for guest OSes installed via LVM or raw disk devices, running
# os-prober can cause damage to those guest OSes as it mounts
# filesystems to look for things.
#GRUB_DISABLE_OS_PROBER=false

# Uncomment to enable BadRAM filtering, modify to suit your needs
# This works with Linux (no patch required) and with any kernel that obtains
# the memory map information from GRUB (GNU Mach, kernel of FreeBSD ...)
#GRUB_BADRAM="0x01234567,0xfefefefe,0x89abcdef,0xefefefef"

# Uncomment to disable graphical terminal
#GRUB_TERMINAL=console

# The resolution used on graphical terminal
# note that you can use only modes which your graphic card supports via VBE
# you can see them in real GRUB with the command 'vbeinfo'
#GRUB_GFXMODE=640x480

# Uncomment if you don't want GRUB to pass "root=UUID=xxx" parameter to Linux
#GRUB_DISABLE_LINUX_UUID=true

# Uncomment to disable generation of recovery mode menu entries
#GRUB_DISABLE_RECOVERY="true"

Help      Write Out  Where Is   Cut        Execute    Location   Undo      Set Mark   To Bracket
Exit      Read File  Replace    Paste      Justify    Go To Line Redo      Copy       Where Was
```

3. Run update-grub to update the GRUB bootloader.

```
root@test:/home/test/0F_Deploy# nano /etc/default/grub
root@test:/home/test/0F_Deploy# update-grub
Sourcing file '/etc/default/grub'
Generating grub configuration file ...
Found linux image: /boot/vmlinuz-6.8.0-134-generic
Found initrd image: /boot/initrd.img-6.8.0-134-generic
Found linux image: /boot/vmlinuz-6.8.0-31-generic
Found initrd image: /boot/initrd.img-6.8.0-31-generic
Warning: os-prober will not be executed to detect other bootable partitions.
Systems on them will not be added to the GRUB boot configuration.
Check GRUB_DISABLE_OS_PROBER documentation entry.
Adding boot menu entry for UEFI Firmware Settings ...
done
root@test:/home/test/0F_Deploy#
```

4. Verify C-states configuration takes effect.

```
#cat /sys/devices/system/cpu/cpu0/cpuidle/state*/name
```

```
#cat /sys/devices/system/cpu/cpu0/cpuidle/state*/disable
```

```
root@test:/home/test# cat /sys/devices/system/cpu/cpu0/cpuidle/state*/name
POLL
C1
root@test:/home/test# cat /sys/devices/system/cpu/cpu0/cpuidle/state*/disable
0
0
root@test:/home/test#
```

3.3.3. Switch to latency-performance mode

1. Install the tuned tool.

#apt install tuned

```
root@test:/home/test# apt install tuned
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
tuned is already the newest version (2.21.0-1ubuntu2).
0 upgraded, 0 newly installed, 0 to remove and 152 not upgraded.
root@test:/home/test#
```

2. Switch to latency-performance mode.

#tuned-adm profile latency-performance

```
root@test:/home/test# tuned-adm profile latency-performance
root@test:/home/test#
```

3. Confirm that the latency-performance mode is set successfully.

#tuned-adm active

```
root@test:/home/test# tuned-adm active
Current active profile: latency-performance
root@test:/home/test#
```

3.3.4. Set up network card port information

1. Check Mellanox network card port information.

#ip link show

```
root@test:/home/test# ip link show
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN mode DEFAULT group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
2: eno8303: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc mq state UP mode DEFAULT group default qlen 1000
    link/ether c8:4b:d6:f8:0e:ea brd ff:ff:ff:ff:ff:ff
    altname enp1s0f0
3: eno8403: <NO-CARRIER,BROADCAST,MULTICAST,UP> mtu 1500 qdisc mq state DOWN mode DEFAULT group default qlen 1000
    link/ether c8:4b:d6:f8:0e:eb brd ff:ff:ff:ff:ff:ff
    altname enp1s0f1
4: enp13s0f0np0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 4500 qdisc mq state UP mode DEFAULT group default qlen 1000
    link/ether e8:9e:49:7c:0b:24 brd ff:ff:ff:ff:ff:ff
5: enp13s0f1np1: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 4500 qdisc mq state UP mode DEFAULT group default qlen 1000
    link/ether e8:9e:49:7c:0b:25 brd ff:ff:ff:ff:ff:ff
root@test:/home/test#
```

2. Edit Mellanox NIC port information.

#nano /etc/netplan/01-netconfig.yaml

Edit the text in the image below:

```
GNU nano 7.2 /etc/netplan/01-netconfig.yaml
network:
  version: 2
  renderer: networkd
  ethernets:
    enp13s0f0np0:
      addresses:
        - 192.168.2.20/24
      mtu: 4500
    enp13s0f1np1:
      addresses:
        - 192.168.3.20/24
      mtu: 4500
```

4. FIO Test

4.1. Connect the SSDs

Note: NQN, target IP address, service port and queue count in above example are for reference only. Please apply parameters according to your real deployment.

1. NVMe-oF Controller-Port1: Connect to the test storage at IP 192.168.2.10

```
#nvme connect -t rdma -n nqn.2015-09.com.wdc:nvme.1 -a 192.168.2.10 -s 4420 --nr-io-queues=16
```

```
#nvme connect -t rdma -n nqn.2015-09.com.wdc:nvme.2 -a 192.168.2.10 -s 4420 --nr-io-queues=16
```

```
#nvme connect -t rdma -n nqn.2015-09.com.wdc:nvme.3 -a 192.168.2.10 -s 4420 --nr-io-queues=16
```

```
#nvme connect -t rdma -n nqn.2015-09.com.wdc:nvme.4 -a 192.168.2.10 -s 4420 --nr-io-queues=16
```

```
#nvme connect -t rdma -n nqn.2015-09.com.wdc:nvme.5 -a 192.168.2.10 -s 4420 --nr-io-queues=16
```

```
#nvme connect -t rdma -n nqn.2015-09.com.wdc:nvme.6 -a 192.168.2.10 -s 4420 --nr-io-queues=16
```

```
#nvme connect -t rdma -n nqn.2015-09.com.wdc:nvme.7 -a 192.168.2.10 -s 4420 --nr-io-queues=16
```

```
#nvme connect -t rdma -n nqn.2015-09.com.wdc:nvme.8 -a 192.168.2.10 -s 4420 --nr-io-queues=16
```

```
#nvme connect -t rdma -n nqn.2015-09.com.wdc:nvme.9 -a 192.168.2.10 -s 4420 --nr-io-queues=16
```

```
#nvme connect -t rdma -n nqn.2015-09.com.wdc:nvme.10 -a 192.168.2.10 -s 4420 --nr-io-queues=16
```

```
#nvme connect -t rdma -n nqn.2015-09.com.wdc:nvme.11 -a 192.168.2.10 -s 4420 --nr-io-queues=16
```

```
#nvme connect -t rdma -n nqn.2015-09.com.wdc:nvme.12 -a 192.168.2.10 -s 4420 --nr-io-queues=16
```

```
root@test:/home/test# nvme connect -t rdma -n nqn.2015-09.com.wdc:nvme.1 -a 192.168.2.10 -s 4420 --nr-io-queues=16
nvme connect -t rdma -n nqn.2015-09.com.wdc:nvme.2 -a 192.168.2.10 -s 4420 --nr-io-queues=16
nvme connect -t rdma -n nqn.2015-09.com.wdc:nvme.3 -a 192.168.2.10 -s 4420 --nr-io-queues=16
nvme connect -t rdma -n nqn.2015-09.com.wdc:nvme.4 -a 192.168.2.10 -s 4420 --nr-io-queues=16
nvme connect -t rdma -n nqn.2015-09.com.wdc:nvme.5 -a 192.168.2.10 -s 4420 --nr-io-queues=16
nvme connect -t rdma -n nqn.2015-09.com.wdc:nvme.6 -a 192.168.2.10 -s 4420 --nr-io-queues=16
nvme connect -t rdma -n nqn.2015-09.com.wdc:nvme.7 -a 192.168.2.10 -s 4420 --nr-io-queues=16
nvme connect -t rdma -n nqn.2015-09.com.wdc:nvme.8 -a 192.168.2.10 -s 4420 --nr-io-queues=16
nvme connect -t rdma -n nqn.2015-09.com.wdc:nvme.9 -a 192.168.2.10 -s 4420 --nr-io-queues=16
nvme connect -t rdma -n nqn.2015-09.com.wdc:nvme.10 -a 192.168.2.10 -s 4420 --nr-io-queues=16
nvme connect -t rdma -n nqn.2015-09.com.wdc:nvme.11 -a 192.168.2.10 -s 4420 --nr-io-queues=16
nvme connect -t rdma -n nqn.2015-09.com.wdc:nvme.12 -a 192.168.2.10 -s 4420 --nr-io-queues=16
connecting to device: nvme1
connecting to device: nvme2
connecting to device: nvme3
connecting to device: nvme4
connecting to device: nvme5
connecting to device: nvme6
connecting to device: nvme7
connecting to device: nvme8
connecting to device: nvme9
connecting to device: nvme10
connecting to device: nvme11
connecting to device: nvme12
root@test:/home/test#
```

2. NVMe-oF Controller-Port2: Connect to the test storage at IP 192.168.3.10


```
#nvme connect -t rdma -n nqn.2015-09.com.wdc:nvme.13 -a 192.168.3.10 -s 4420 --nr-io-queues=16

#nvme connect -t rdma -n nqn.2015-09.com.wdc:nvme.14 -a 192.168.3.10 -s 4420 --nr-io-queues=16

#nvme connect -t rdma -n nqn.2015-09.com.wdc:nvme.15 -a 192.168.3.10 -s 4420 --nr-io-queues=16

#nvme connect -t rdma -n nqn.2015-09.com.wdc:nvme.16 -a 192.168.3.10 -s 4420 --nr-io-queues=16

#nvme connect -t rdma -n nqn.2015-09.com.wdc:nvme.17 -a 192.168.3.10 -s 4420 --nr-io-queues=16

#nvme connect -t rdma -n nqn.2015-09.com.wdc:nvme.18 -a 192.168.3.10 -s 4420 --nr-io-queues=16

#nvme connect -t rdma -n nqn.2015-09.com.wdc:nvme.19 -a 192.168.3.10 -s 4420 --nr-io-queues=16

#nvme connect -t rdma -n nqn.2015-09.com.wdc:nvme.20 -a 192.168.3.10 -s 4420 --nr-io-queues=16

#nvme connect -t rdma -n nqn.2015-09.com.wdc:nvme.21 -a 192.168.3.10 -s 4420 --nr-io-queues=16

#nvme connect -t rdma -n nqn.2015-09.com.wdc:nvme.22 -a 192.168.3.10 -s 4420 --nr-io-queues=16

#nvme connect -t rdma -n nqn.2015-09.com.wdc:nvme.23 -a 192.168.3.10 -s 4420 --nr-io-queues=16

#nvme connect -t rdma -n nqn.2015-09.com.wdc:nvme.24 -a 192.168.3.10 -s 4420 --nr-io-queues=16
```

```
root@test:/home/test# nvme connect -t rdma -n nqn.2015-09.com.wdc:nvme.13 -a 192.168.3.10 -s 4420 --nr-io-queues=16
nvme connect -t rdma -n nqn.2015-09.com.wdc:nvme.14 -a 192.168.3.10 -s 4420 --nr-io-queues=16
nvme connect -t rdma -n nqn.2015-09.com.wdc:nvme.15 -a 192.168.3.10 -s 4420 --nr-io-queues=16
nvme connect -t rdma -n nqn.2015-09.com.wdc:nvme.16 -a 192.168.3.10 -s 4420 --nr-io-queues=16
nvme connect -t rdma -n nqn.2015-09.com.wdc:nvme.17 -a 192.168.3.10 -s 4420 --nr-io-queues=16
nvme connect -t rdma -n nqn.2015-09.com.wdc:nvme.18 -a 192.168.3.10 -s 4420 --nr-io-queues=16
nvme connect -t rdma -n nqn.2015-09.com.wdc:nvme.19 -a 192.168.3.10 -s 4420 --nr-io-queues=16
nvme connect -t rdma -n nqn.2015-09.com.wdc:nvme.20 -a 192.168.3.10 -s 4420 --nr-io-queues=16
nvme connect -t rdma -n nqn.2015-09.com.wdc:nvme.21 -a 192.168.3.10 -s 4420 --nr-io-queues=16
nvme connect -t rdma -n nqn.2015-09.com.wdc:nvme.22 -a 192.168.3.10 -s 4420 --nr-io-queues=16
nvme connect -t rdma -n nqn.2015-09.com.wdc:nvme.23 -a 192.168.3.10 -s 4420 --nr-io-queues=16
nvme connect -t rdma -n nqn.2015-09.com.wdc:nvme.24 -a 192.168.3.10 -s 4420 --nr-io-queues=16
connecting to device: nvme13
connecting to device: nvme14
connecting to device: nvme15
connecting to device: nvme16
connecting to device: nvme17
connecting to device: nvme18
connecting to device: nvme19
connecting to device: nvme20
connecting to device: nvme21
connecting to device: nvme22
connecting to device: nvme23
connecting to device: nvme24
```

3. List NVMe devices with verbose details.

```
#nvme list -v
```

```
root@test:/home/test# nvme list -v
Subsystem      Subsystem-NQN          Controllers
-----
nvme-subsys0   nqn.2019-10.com.kioxia:K1012-KC071RLK986G:81E0A00KTL7  nvme0
nvme-subsys1   nqn.2015-09.com.wdc:nvme.1  nvme1
nvme-subsys10  nqn.2015-09.com.wdc:nvme.10  nvme10
nvme-subsys11  nqn.2015-09.com.wdc:nvme.11  nvme11
nvme-subsys12  nqn.2015-09.com.wdc:nvme.12  nvme12
nvme-subsys13  nqn.2015-09.com.wdc:nvme.13  nvme13
nvme-subsys14  nqn.2015-09.com.wdc:nvme.14  nvme14
nvme-subsys15  nqn.2015-09.com.wdc:nvme.15  nvme15
nvme-subsys16  nqn.2015-09.com.wdc:nvme.16  nvme16
nvme-subsys17  nqn.2015-09.com.wdc:nvme.17  nvme17
nvme-subsys18  nqn.2015-09.com.wdc:nvme.18  nvme18
nvme-subsys19  nqn.2015-09.com.wdc:nvme.19  nvme19
nvme-subsys2   nqn.2015-09.com.wdc:nvme.2  nvme2
nvme-subsys20  nqn.2015-09.com.wdc:nvme.20  nvme20
nvme-subsys21  nqn.2015-09.com.wdc:nvme.21  nvme21
nvme-subsys22  nqn.2015-09.com.wdc:nvme.22  nvme22
nvme-subsys23  nqn.2015-09.com.wdc:nvme.23  nvme23
nvme-subsys24  nqn.2015-09.com.wdc:nvme.24  nvme24
nvme-subsys3   nqn.2015-09.com.wdc:nvme.3  nvme3
nvme-subsys4   nqn.2015-09.com.wdc:nvme.4  nvme4
nvme-subsys5   nqn.2015-09.com.wdc:nvme.5  nvme5
nvme-subsys6   nqn.2015-09.com.wdc:nvme.6  nvme6
nvme-subsys7   nqn.2015-09.com.wdc:nvme.7  nvme7
nvme-subsys8   nqn.2015-09.com.wdc:nvme.8  nvme8
nvme-subsys9   nqn.2015-09.com.wdc:nvme.9  nvme9
```


Device	SN	MN	FR	TxPort	Address	Slot	Subsystem	Namespaces
nvme0	81E0A0K07	KI0X1A KC071RU9960	0104	pcie	0000:82:00:0		nvme-subsys0	nvme0n1
nvme10	24180H900082	WUSSEC1BIESP777	RC910A00	rdma	traddr=192.168.2.10, trsvcid=4420		nvme-subsys10	nvme10n1
nvme11	24180H900082	WUSSEC1BIESP777	RC910A00	rdma	traddr=192.168.2.10, trsvcid=4420		nvme-subsys11	nvme11n1
nvme12	24180H900081	WUSSEC1BIESP777	RC910A00	rdma	traddr=192.168.2.10, trsvcid=4420		nvme-subsys12	nvme12n1
nvme13	24180H900069	WUSSEC1BIESP777	RC910A00	rdma	traddr=192.168.3.10, trsvcid=4420		nvme-subsys13	nvme13n1
nvme14	24180H900066	WUSSEC1BIESP777	RC910A00	rdma	traddr=192.168.3.10, trsvcid=4420		nvme-subsys14	nvme14n1
nvme15	24180H900017	WUSSEC1BIESP777	RC910A00	rdma	traddr=192.168.3.10, trsvcid=4420		nvme-subsys15	nvme15n1
nvme16	24180H900045	WUSSEC1BIESP777	RC910A00	rdma	traddr=192.168.3.10, trsvcid=4420		nvme-subsys16	nvme16n1
nvme17	24180H900039	WUSSEC1BIESP777	RC910A00	rdma	traddr=192.168.3.10, trsvcid=4420		nvme-subsys17	nvme17n1
nvme18	24180H900040	WUSSEC1BIESP777	RC910A00	rdma	traddr=192.168.3.10, trsvcid=4420		nvme-subsys18	nvme18n1
nvme19	24180L900007	WUSSEC1BIESP777	RC910A00	rdma	traddr=192.168.3.10, trsvcid=4420		nvme-subsys19	nvme19n1
nvme20	24180H900062	WUSSEC1BIESP777	RC910A00	rdma	traddr=192.168.2.10, trsvcid=4420		nvme-subsys20	nvme20n1
nvme21	24180H900043	WUSSEC1BIESP777	RC910A00	rdma	traddr=192.168.3.10, trsvcid=4420		nvme-subsys21	nvme21n1
nvme22	24180L900079	WUSSEC1BIESP777	RC910A00	rdma	traddr=192.168.3.10, trsvcid=4420		nvme-subsys22	nvme22n1
nvme23	24180H900053	WUSSEC1BIESP777	RC910A00	rdma	traddr=192.168.3.10, trsvcid=4420		nvme-subsys23	nvme23n1
nvme24	24180H900036	WUSSEC1BIESP777	RC910A00	rdma	traddr=192.168.3.10, trsvcid=4420		nvme-subsys24	nvme24n1
nvme3	24180L900032	WUSSEC1BIESP777	RC910A00	rdma	traddr=192.168.2.10, trsvcid=4420		nvme-subsys3	nvme3n1
nvme4	24180H900082	WUSSEC1BIESP777	RC910A00	rdma	traddr=192.168.2.10, trsvcid=4420		nvme-subsys4	nvme4n1
nvme5	24180H900021	WUSSEC1BIESP777	RC910A00	rdma	traddr=192.168.2.10, trsvcid=4420		nvme-subsys5	nvme5n1
nvme6	24180H900019	WUSSEC1BIESP777	RC910A00	rdma	traddr=192.168.2.10, trsvcid=4420		nvme-subsys6	nvme6n1
nvme7	24180H900075	WUSSEC1BIESP777	RC910A00	rdma	traddr=192.168.2.10, trsvcid=4420		nvme-subsys7	nvme7n1
nvme8	24180L900036	WUSSEC1BIESP777	RC910A00	rdma	traddr=192.168.2.10, trsvcid=4420		nvme-subsys8	nvme8n1
nvme9	24180H900046	WUSSEC1BIESP777	RC910A00	rdma	traddr=192.168.2.10, trsvcid=4420		nvme-subsys9	nvme9n1

Device	Generic	NSID	Usage	Format	Controllers
/dev/nvme0n1	/dev/ng0n1	0x1	5.96 GB / 960.20 GB	512 B + 0 B	nvme0
/dev/nvme10n1	/dev/ng10n1	0x1	30.72 TB / 245.76 TB	4 KiB + 0 B	nvme10
/dev/nvme11n1	/dev/ng11n1	0x1	30.72 TB / 245.76 TB	4 KiB + 0 B	nvme11
/dev/nvme12n1	/dev/ng12n1	0x1	30.72 TB / 245.76 TB	4 KiB + 0 B	nvme12
/dev/nvme13n1	/dev/ng13n1	0x1	30.72 TB / 245.76 TB	4 KiB + 0 B	nvme13
/dev/nvme14n1	/dev/ng14n1	0x1	30.72 TB / 245.76 TB	4 KiB + 0 B	nvme14
/dev/nvme15n1	/dev/ng15n1	0x1	30.72 TB / 245.76 TB	4 KiB + 0 B	nvme15
/dev/nvme16n1	/dev/ng16n1	0x1	30.72 TB / 245.76 TB	4 KiB + 0 B	nvme16
/dev/nvme17n1	/dev/ng17n1	0x1	30.72 TB / 245.76 TB	4 KiB + 0 B	nvme17
/dev/nvme18n1	/dev/ng18n1	0x1	30.72 TB / 245.76 TB	4 KiB + 0 B	nvme18
/dev/nvme19n1	/dev/ng19n1	0x1	30.72 TB / 245.76 TB	4 KiB + 0 B	nvme19
/dev/nvme20n1	/dev/ng20n1	0x1	30.72 TB / 245.76 TB	4 KiB + 0 B	nvme20
/dev/nvme21n1	/dev/ng21n1	0x1	30.72 TB / 245.76 TB	4 KiB + 0 B	nvme21
/dev/nvme22n1	/dev/ng22n1	0x1	30.72 TB / 245.76 TB	4 KiB + 0 B	nvme22
/dev/nvme23n1	/dev/ng23n1	0x1	30.72 TB / 245.76 TB	4 KiB + 0 B	nvme23
/dev/nvme24n1	/dev/ng24n1	0x1	30.72 TB / 245.76 TB	4 KiB + 0 B	nvme24
/dev/nvme2n1	/dev/ng2n1	0x1	30.72 TB / 245.76 TB	4 KiB + 0 B	nvme2
/dev/nvme3n1	/dev/ng3n1	0x1	30.72 TB / 245.76 TB	4 KiB + 0 B	nvme3
/dev/nvme4n1	/dev/ng4n1	0x1	30.72 TB / 245.76 TB	4 KiB + 0 B	nvme4
/dev/nvme5n1	/dev/ng5n1	0x1	30.72 TB / 245.76 TB	4 KiB + 0 B	nvme5
/dev/nvme6n1	/dev/ng6n1	0x1	30.72 TB / 245.76 TB	4 KiB + 0 B	nvme6
/dev/nvme7n1	/dev/ng7n1	0x1	30.72 TB / 245.76 TB	4 KiB + 0 B	nvme7
/dev/nvme8n1	/dev/ng8n1	0x1	30.72 TB / 245.76 TB	4 KiB + 0 B	nvme8
/dev/nvme9n1	/dev/ng9n1	0x1	30.72 TB / 245.76 TB	4 KiB + 0 B	nvme9

4.2. Format the SSDs

Command to format target device as ext4 filesystem.

#mkfs.ext4 -F /dev/nvmeXnY

Example: **mkfs.ext4 -F /dev/nvme1n1**

```

root@test:/home/test# mkfs.ext4 -F /dev/nvme1n1
mke2fs 1.47.0 (5-Feb-2023)
Discarding device blocks: done
Creating filesystem with 7500000000 4k blocks and 468750336 inodes
Filesystem UUID: 093e587c-db1e-4618-9457-7b2cad2b4b0f
Superblock backups stored on blocks:
    32768, 98304, 163840, 229376, 294912, 819200, 884736, 1605632, 2654208,
    4096000, 7962624, 11239424, 20480000, 23887872, 71663616, 78675968,
    102400000, 214990848, 512000000, 550731776, 644972544, 1934917632,
    2560000000, 3855122432, 5804752896

Allocating group tables: done
Writing inode tables: done
Creating journal (262144 blocks): done
Writing superblocks and filesystem accounting information: done

root@test:/home/test#

```

Note: nvmeXnY: Target NVMe raw disk; X=controller ID, Y=namespace ID.

4.3. FIO Test Scripts

4.3.1. Single-port test scripts

● NVMe-oF Controller-Port 1

■ Sequential Read

```
# fio --time_based --runtime=300 --loops=1 --direct=1 --rw=read --
unified_rw_reporting=0 --refill_buffers --random_generator=tausworthe64 --
norandommap --randrepeat=0 --ioengine=libaio --bs=131072 --iodepth=16 --numjobs=1
--group_reporting --size=100% --offset=0% --name="Job 1" --filename=/dev/nvme1n1 --
--name="Job 2" --filename=/dev/nvme2n1 --name="Job 3" --filename=/dev/nvme3n1 --
--name="Job 4" --filename=/dev/nvme4n1 --name="Job 5" --filename=/dev/nvme5n1 --
--name="Job 6" --filename=/dev/nvme6n1 --name="Job 7" --filename=/dev/nvme7n1 --
--name="Job 8" --filename=/dev/nvme8n1 --name="Job 9" --filename=/dev/nvme9n1 --
--name="Job 10" --filename=/dev/nvme10n1 --name="Job 11" --
filename=/dev/nvme11n1 --name="Job 12" --filename=/dev/nvme12n1
```

```
root@test:/home/test# fio --time_based --runtime=300 --loops=1 --direct=1 --rw=read --unified_rw_reporting=0 --refill_buffers --random_generator=tausworthe64
--norandommap --randrepeat=0 --ioengine=libaio --bs=131072 --iodepth=16 --numjobs=1 --group_reporting --size=100% --offset=0% --name="Job 1" --filename=/dev/nvme1n1 --
--name="Job 2" --filename=/dev/nvme2n1 --name="Job 3" --filename=/dev/nvme3n1 --name="Job 4" --filename=/dev/nvme4n1 --name="Job 5" --filename=/dev/nvme5n1 --
--name="Job 6" --filename=/dev/nvme6n1 --name="Job 7" --filename=/dev/nvme7n1 --name="Job 8" --filename=/dev/nvme8n1 --name="Job 9" --filename=/dev/nvme9n1 --
--name="Job 10" --filename=/dev/nvme10n1 --name="Job 11" --filename=/dev/nvme11n1 --name="Job 12" --filename=/dev/nvme12n1
Job 1: (g0): rw=read, bs=(R) 128KiB-128KiB, (W) 128KiB-128KiB, (T) 128KiB-128KiB, ioengine=libaio, iodepth=16
Job 2: (g0): rw=read, bs=(R) 128KiB-128KiB, (W) 128KiB-128KiB, (T) 128KiB-128KiB, ioengine=libaio, iodepth=16
Job 3: (g0): rw=read, bs=(R) 128KiB-128KiB, (W) 128KiB-128KiB, (T) 128KiB-128KiB, ioengine=libaio, iodepth=16
Job 4: (g0): rw=read, bs=(R) 128KiB-128KiB, (W) 128KiB-128KiB, (T) 128KiB-128KiB, ioengine=libaio, iodepth=16
Job 5: (g0): rw=read, bs=(R) 128KiB-128KiB, (W) 128KiB-128KiB, (T) 128KiB-128KiB, ioengine=libaio, iodepth=16
Job 6: (g0): rw=read, bs=(R) 128KiB-128KiB, (W) 128KiB-128KiB, (T) 128KiB-128KiB, ioengine=libaio, iodepth=16
Job 7: (g0): rw=read, bs=(R) 128KiB-128KiB, (W) 128KiB-128KiB, (T) 128KiB-128KiB, ioengine=libaio, iodepth=16
Job 8: (g0): rw=read, bs=(R) 128KiB-128KiB, (W) 128KiB-128KiB, (T) 128KiB-128KiB, ioengine=libaio, iodepth=16
Job 9: (g0): rw=read, bs=(R) 128KiB-128KiB, (W) 128KiB-128KiB, (T) 128KiB-128KiB, ioengine=libaio, iodepth=16
Job 10: (g0): rw=read, bs=(R) 128KiB-128KiB, (W) 128KiB-128KiB, (T) 128KiB-128KiB, ioengine=libaio, iodepth=16
Job 11: (g0): rw=read, bs=(R) 128KiB-128KiB, (W) 128KiB-128KiB, (T) 128KiB-128KiB, ioengine=libaio, iodepth=16
Job 12: (g0): rw=read, bs=(R) 128KiB-128KiB, (W) 128KiB-128KiB, (T) 128KiB-128KiB, ioengine=libaio, iodepth=16
fio-3.36
Starting 12 processes
Jobs: 12 (f=12): [R(12)][21.0%][r=11.4GiB/s][w=93.1k IOPS][eta 03m:57s]
```

■ Sequential Write

```
# fio --time_based --runtime=300 --loops=1 --direct=1 --rw=write --
unified_rw_reporting=0 --refill_buffers --random_generator=tausworthe64 --
norandommap --randrepeat=0 --ioengine=libaio --bs=131072 --iodepth=16 --numjobs=1
--group_reporting --size=100% --offset=0% --name="Job 1" --filename=/dev/nvme1n1 --
--name="Job 2" --filename=/dev/nvme2n1 --name="Job 3" --filename=/dev/nvme3n1 --
--name="Job 4" --filename=/dev/nvme4n1 --name="Job 5" --filename=/dev/nvme5n1 --
--name="Job 6" --filename=/dev/nvme6n1 --name="Job 7" --filename=/dev/nvme7n1 --
--name="Job 8" --filename=/dev/nvme8n1 --name="Job 9" --filename=/dev/nvme9n1 --
--name="Job 10" --filename=/dev/nvme10n1 --name="Job 11" --
filename=/dev/nvme11n1 --name="Job 12" --filename=/dev/nvme12n1
```

```
root@test:/home/test# fio --time_based --runtime=300 --loops=1 --direct=1 --rw=write --unified_rw_reporting=0 --refill_buffers --random_generator=tausworthe64
--norandommap --randrepeat=0 --ioengine=libaio --bs=131072 --iodepth=16 --numjobs=1 --group_reporting --size=100% --offset=0% --name="Job 1" --filename=/dev/nvme1n1 --
--name="Job 2" --filename=/dev/nvme2n1 --name="Job 3" --filename=/dev/nvme3n1 --name="Job 4" --filename=/dev/nvme4n1 --name="Job 5" --filename=/dev/nvme5n1 --
--name="Job 6" --filename=/dev/nvme6n1 --name="Job 7" --filename=/dev/nvme7n1 --name="Job 8" --filename=/dev/nvme8n1 --name="Job 9" --filename=/dev/nvme9n1 --
--name="Job 10" --filename=/dev/nvme10n1 --name="Job 11" --filename=/dev/nvme11n1 --name="Job 12" --filename=/dev/nvme12n1
Job 1: (g0): rw=write, bs=(R) 128KiB-128KiB, (W) 128KiB-128KiB, (T) 128KiB-128KiB, ioengine=libaio, iodepth=16
Job 2: (g0): rw=write, bs=(R) 128KiB-128KiB, (W) 128KiB-128KiB, (T) 128KiB-128KiB, ioengine=libaio, iodepth=16
Job 3: (g0): rw=write, bs=(R) 128KiB-128KiB, (W) 128KiB-128KiB, (T) 128KiB-128KiB, ioengine=libaio, iodepth=16
Job 4: (g0): rw=write, bs=(R) 128KiB-128KiB, (W) 128KiB-128KiB, (T) 128KiB-128KiB, ioengine=libaio, iodepth=16
Job 5: (g0): rw=write, bs=(R) 128KiB-128KiB, (W) 128KiB-128KiB, (T) 128KiB-128KiB, ioengine=libaio, iodepth=16
Job 6: (g0): rw=write, bs=(R) 128KiB-128KiB, (W) 128KiB-128KiB, (T) 128KiB-128KiB, ioengine=libaio, iodepth=16
Job 7: (g0): rw=write, bs=(R) 128KiB-128KiB, (W) 128KiB-128KiB, (T) 128KiB-128KiB, ioengine=libaio, iodepth=16
Job 8: (g0): rw=write, bs=(R) 128KiB-128KiB, (W) 128KiB-128KiB, (T) 128KiB-128KiB, ioengine=libaio, iodepth=16
Job 9: (g0): rw=write, bs=(R) 128KiB-128KiB, (W) 128KiB-128KiB, (T) 128KiB-128KiB, ioengine=libaio, iodepth=16
Job 10: (g0): rw=write, bs=(R) 128KiB-128KiB, (W) 128KiB-128KiB, (T) 128KiB-128KiB, ioengine=libaio, iodepth=16
Job 11: (g0): rw=write, bs=(R) 128KiB-128KiB, (W) 128KiB-128KiB, (T) 128KiB-128KiB, ioengine=libaio, iodepth=16
Job 12: (g0): rw=write, bs=(R) 128KiB-128KiB, (W) 128KiB-128KiB, (T) 128KiB-128KiB, ioengine=libaio, iodepth=16
fio-3.36
Starting 12 processes
Jobs: 12 (f=12): [W(12)][21.7%][w=11.4GiB/s][r=93.4k IOPS][eta 03m:55s]
```

● NVMe-oF Controller-Port 2

■ Sequential Read

```
# fio --time_based --runtime=300 --loops=1 --direct=1 --rw=read --
unified_rw_reporting=0 --refill_buffers --random_generator=tausworthe64 --
```

```

norandommap --randrepeat=0 --ioengine=libaio --bs=131072 --iodepth=16 --numjobs=1
--group_reporting --size=100% --offset=0% --name="Job 13" --
filename=/dev/nvme13n1 --name="Job 14" --filename=/dev/nvme14n1 --name="Job
15" --filename=/dev/nvme15n1 --name="Job 16" --filename=/dev/nvme16n1 --
name="Job 17" --filename=/dev/nvme17n1 --name="Job 18" --
filename=/dev/nvme18n1 --name="Job 19" --filename=/dev/nvme19n1 --name="Job
20" --filename=/dev/nvme20n1 --name="Job 21" --filename=/dev/nvme21n1 --
name="Job 22" --filename=/dev/nvme22n1 --name="Job 23" --
filename=/dev/nvme23n1 --name="Job 24" --filename=/dev/nvme24n1

```

```

root@test:/home/test# fio --time_based --runtime=300 --loops=1 --direct=1 --reread --unified_rw_reporting=0 --refill_buffers --random_generator=tausworthe64
--norandommap --randrepeat=0 --ioengine=libaio --bs=131072 --iodepth=16 --numjobs=1 --group_reporting --size=100% --offset=0% --name="Job 13" --filename=
/dev/nvme13n1 --name="Job 14" --filename=/dev/nvme14n1 --name="Job 15" --filename=/dev/nvme15n1 --name="Job 16" --filename=/dev/nvme16n1 --name="Job 17" --f
ilename=/dev/nvme17n1 --name="Job 18" --filename=/dev/nvme18n1 --name="Job 19" --filename=/dev/nvme19n1 --name="Job 20" --filename=/dev/nvme20n1 --name="Job
21" --filename=/dev/nvme21n1 --name="Job 22" --filename=/dev/nvme22n1 --name="Job 23" --filename=/dev/nvme23n1 --name="Job 24" --filename=/dev/nvme24n1
Job 13: (g0): rwreread, bs=(R) 128KiB-128KiB, (W) 128KiB-128KiB, (T) 128KiB-128KiB, ioengine=libaio, iodepth=16
Job 14: (g0): rwreread, bs=(R) 128KiB-128KiB, (W) 128KiB-128KiB, (T) 128KiB-128KiB, ioengine=libaio, iodepth=16
Job 15: (g0): rwreread, bs=(R) 128KiB-128KiB, (W) 128KiB-128KiB, (T) 128KiB-128KiB, ioengine=libaio, iodepth=16
Job 16: (g0): rwreread, bs=(R) 128KiB-128KiB, (W) 128KiB-128KiB, (T) 128KiB-128KiB, ioengine=libaio, iodepth=16
Job 17: (g0): rwreread, bs=(R) 128KiB-128KiB, (W) 128KiB-128KiB, (T) 128KiB-128KiB, ioengine=libaio, iodepth=16
Job 18: (g0): rwreread, bs=(R) 128KiB-128KiB, (W) 128KiB-128KiB, (T) 128KiB-128KiB, ioengine=libaio, iodepth=16
Job 19: (g0): rwreread, bs=(R) 128KiB-128KiB, (W) 128KiB-128KiB, (T) 128KiB-128KiB, ioengine=libaio, iodepth=16
Job 20: (g0): rwreread, bs=(R) 128KiB-128KiB, (W) 128KiB-128KiB, (T) 128KiB-128KiB, ioengine=libaio, iodepth=16
Job 21: (g0): rwreread, bs=(R) 128KiB-128KiB, (W) 128KiB-128KiB, (T) 128KiB-128KiB, ioengine=libaio, iodepth=16
Job 22: (g0): rwreread, bs=(R) 128KiB-128KiB, (W) 128KiB-128KiB, (T) 128KiB-128KiB, ioengine=libaio, iodepth=16
Job 23: (g0): rwreread, bs=(R) 128KiB-128KiB, (W) 128KiB-128KiB, (T) 128KiB-128KiB, ioengine=libaio, iodepth=16
Job 24: (g0): rwreread, bs=(R) 128KiB-128KiB, (W) 128KiB-128KiB, (T) 128KiB-128KiB, ioengine=libaio, iodepth=16
fio-3.36
Starting 12 processes
[obs: 12 (f=12): [R(12)]:[20.7%][r=11.46iB/s][r=93.6k IOPS][eta 03m:58s]

```

■ Sequential Write

```

# fio --time_based --runtime=300 --loops=1 --direct=1 --rw=write --
unified_rw_reporting=0 --refill_buffers --random_generator=tausworthe64 --
norandommap --randrepeat=0 --ioengine=libaio --bs=131072 --iodepth=16 --numjobs=1
--group_reporting --size=100% --offset=0% --name="Job 13" --
filename=/dev/nvme13n1 --name="Job 14" --filename=/dev/nvme14n1 --name="Job
15" --filename=/dev/nvme15n1 --name="Job 16" --filename=/dev/nvme16n1 --
name="Job 17" --filename=/dev/nvme17n1 --name="Job 18" --
filename=/dev/nvme18n1 --name="Job 19" --filename=/dev/nvme19n1 --name="Job
20" --filename=/dev/nvme20n1 --name="Job 21" --filename=/dev/nvme21n1 --
name="Job 22" --filename=/dev/nvme22n1 --name="Job 23" --
filename=/dev/nvme23n1 --name="Job 24" --filename=/dev/nvme24n1

```

```

root@test:/home/test# fio --time_based --runtime=300 --loops=1 --direct=1 --rwwrite --unified_rw_reporting=0 --refill_buffers --random_generator=tausworthe64
--norandommap --randrepeat=0 --ioengine=libaio --bs=131072 --iodepth=16 --numjobs=1 --group_reporting --size=100% --offset=0% --name="Job 13" --filename=
/dev/nvme13n1 --name="Job 14" --filename=/dev/nvme14n1 --name="Job 15" --filename=/dev/nvme15n1 --name="Job 16" --filename=/dev/nvme16n1 --name="Job 17" --f
ilename=/dev/nvme17n1 --name="Job 18" --filename=/dev/nvme18n1 --name="Job 19" --filename=/dev/nvme19n1 --name="Job 20" --filename=/dev/nvme20n1 --name="Job
21" --filename=/dev/nvme21n1 --name="Job 22" --filename=/dev/nvme22n1 --name="Job 23" --filename=/dev/nvme23n1 --name="Job 24" --filename=/dev/nvme24n1
Job 13: (g0): rwwrite, bs=(R) 128KiB-128KiB, (W) 128KiB-128KiB, (T) 128KiB-128KiB, ioengine=libaio, iodepth=16
Job 14: (g0): rwwrite, bs=(R) 128KiB-128KiB, (W) 128KiB-128KiB, (T) 128KiB-128KiB, ioengine=libaio, iodepth=16
Job 15: (g0): rwwrite, bs=(R) 128KiB-128KiB, (W) 128KiB-128KiB, (T) 128KiB-128KiB, ioengine=libaio, iodepth=16
Job 16: (g0): rwwrite, bs=(R) 128KiB-128KiB, (W) 128KiB-128KiB, (T) 128KiB-128KiB, ioengine=libaio, iodepth=16
Job 17: (g0): rwwrite, bs=(R) 128KiB-128KiB, (W) 128KiB-128KiB, (T) 128KiB-128KiB, ioengine=libaio, iodepth=16
Job 18: (g0): rwwrite, bs=(R) 128KiB-128KiB, (W) 128KiB-128KiB, (T) 128KiB-128KiB, ioengine=libaio, iodepth=16
Job 19: (g0): rwwrite, bs=(R) 128KiB-128KiB, (W) 128KiB-128KiB, (T) 128KiB-128KiB, ioengine=libaio, iodepth=16
Job 20: (g0): rwwrite, bs=(R) 128KiB-128KiB, (W) 128KiB-128KiB, (T) 128KiB-128KiB, ioengine=libaio, iodepth=16
Job 21: (g0): rwwrite, bs=(R) 128KiB-128KiB, (W) 128KiB-128KiB, (T) 128KiB-128KiB, ioengine=libaio, iodepth=16
Job 22: (g0): rwwrite, bs=(R) 128KiB-128KiB, (W) 128KiB-128KiB, (T) 128KiB-128KiB, ioengine=libaio, iodepth=16
Job 23: (g0): rwwrite, bs=(R) 128KiB-128KiB, (W) 128KiB-128KiB, (T) 128KiB-128KiB, ioengine=libaio, iodepth=16
Job 24: (g0): rwwrite, bs=(R) 128KiB-128KiB, (W) 128KiB-128KiB, (T) 128KiB-128KiB, ioengine=libaio, iodepth=16
fio-3.36
Starting 12 processes
[obs: 12 (f=12): [W(12)]:[64.0%][w=11.46iB/s][w=93.4k IOPS][eta 01m:48s]

```

4.3.2. Dual-port test scripts

● Sequential Read

```

#fio --time_based --runtime=300 --loops=1 --direct=1 --rw=read --unified_rw_reporting=0 --
refill_buffers --random_generator=tausworthe64 --norandommap --randrepeat=0 --
ioengine=libaio --bs=131072 --iodepth=16 --numjobs=1 --group_reporting --size=100% --
offset=0% --name="Job 1" --filename=/dev/nvme1n1 --name="Job 2" --
filename=/dev/nvme2n1 --name="Job 3" --filename=/dev/nvme3n1 --name="Job 4" --
filename=/dev/nvme4n1 --name="Job 5" --filename=/dev/nvme5n1 --name="Job 6" --
filename=/dev/nvme6n1 --name="Job 7" --filename=/dev/nvme7n1 --name="Job 8" --
filename=/dev/nvme8n1 --name="Job 9" --filename=/dev/nvme9n1 --name="Job 10" --
filename=/dev/nvme10n1 --name="Job 11" --filename=/dev/nvme11n1 --name="Job 12" --
filename=/dev/nvme12n1 --name="Job 13" --filename=/dev/nvme13n1 --name="Job 14" --

```


[illegible]

```

#no --time_based --runtime=500 --loops=1 --direct=1 --rw --write --unified_rw_reporting=0 --
refill_buffers --random_generator=tausworthe64 --norandommap --randrepeat=0 --
ioengine=libaio --bs=131072 --iodepth=16 --numjobs=1 --group_reporting --size=100% --
offset=0% --name="Job 1" --filename=/dev/nvme1n1 --name="Job 2" --
filename=/dev/nvme2n1 --name="Job 3" --filename=/dev/nvme3n1 --name="Job 4" --
filename=/dev/nvme4n1 --name="Job 5" --filename=/dev/nvme5n1 --name="Job 6" --
filename=/dev/nvme6n1 --name="Job 7" --filename=/dev/nvme7n1 --name="Job 8" --
filename=/dev/nvme8n1 --name="Job 9" --filename=/dev/nvme9n1 --name="Job 10" --
filename=/dev/nvme10n1 --name="Job 11" --filename=/dev/nvme11n1 --name="Job 12" --
filename=/dev/nvme12n1 --name="Job 13" --filename=/dev/nvme13n1 --name="Job 14" --
filename=/dev/nvme14n1 --name="Job 15" --filename=/dev/nvme15n1 --name="Job 16" --
filename=/dev/nvme16n1 --name="Job 17" --filename=/dev/nvme17n1 --name="Job 18" --
filename=/dev/nvme18n1 --name="Job 19" --filename=/dev/nvme19n1 --name="Job 20" --
filename=/dev/nvme20n1 --name="Job 21" --filename=/dev/nvme21n1 --name="Job 22" --
filename=/dev/nvme22n1 --name="Job 23" --filename=/dev/nvme23n1 --name="Job 24" --
filename=/dev/nvme24n1

```

Point Technologies, Inc 15

4.4. Performance Summary

Notes:

During testing, we used 24 SSDs, with 12 drives connected to each port.

One cycle consists of Formatting → Read Test → Write Test.

FIO read workload duration: 5 minutes (300 seconds)

FIO write workload duration: 5 minutes (300 seconds)

4.4.1. Single-port test results

Test No.	Sequential-read Performance value	Sequential-write Performance value
1	12.2 GB/s	12.2 GB/s

4.4.2. Dual-port test results

Test No.	Sequential-read Performance value	Sequential-write Performance value
1	21 GB/s	17.5 GB/s
2	21 GB/s	17.5 GB/s
3	20.9 GB/s	17.1 GB/s