RHEL 9

SAN hosts and cloud clients

NetApp
January 31, 2023

Table of Contents

RHEL 9

NVMe-oF Host Configuration for RHEL 9.1 with ONTAP

Supportability

NVMe over Fabrics or NVMe-oF (including NVMe/FC and NVMe/TCP) is supported with RHEL 9.1 with Asymmetric Namespace Access (ANA) that is required for surviving storage failovers (SFOs) on the ONTAP array. ANA is the asymmetric logical unit access (ALUA) equivalent in the NVMe-oF environment, and is currently implemented with in-kernel NVMe Multipath. This document contains the details for enabling NVMe-oF with in-kernel NVMe Multipath using ANA on RHEL 9.1 and ONTAP as the target.

You can use the configuration settings provided in this document to configure cloud clients connected to Cloud Volumes ONTAP and Amazon FSx for ONTAP.

Features

- RHEL 9.1 includes support for NVMe/TCP in addition to NVMe/FC. The NetApp plugin in the native `nvme-cli` package can display ONTAP details for both NVMe/FC and NVMe/TCP namespaces.
- RHEL 9.1 includes support for in-kernel NVMe multipath for NVMe namespaces enabled by default, without the need for explicit settings.
- RHEL 9.1 supports use of NVMe and SCSI co-existent traffic on the same host on a given HBA adapter, without the explicit `dm-multipath` settings to prevent claiming NVMe namespaces.

Configuration requirements

Refer to the NetApp Interoperability Matrix for accurate details regarding supported configurations.

Enable in-kernel NVMe multipath

Steps

1. Install RHEL 9.1 on the server. After the installation is complete, verify that you are running the specified RHEL 9.1 kernel. See the NetApp Interoperability Matrix for the most current list of supported versions.
2. After the installation is complete, verify that you are running the specified RHEL 9.1 kernel. See the NetApp Interoperability Matrix for the most current list of supported versions.

   Example:

   ```
   # uname -r
    5.14.0-162.6.1.el9_1.x86_64
   ```

3. Install the `nvme-cli` package:

   Example:

```
# rpm -qa|grep nvme-cli
nvme-cli-2.0-4.el9.x86_64
```

4. On the host, check the host NQN string at `/etc/nvme/hostnqn` and verify that it matches the host NQN string for the corresponding subsystem on the ONTAP array. Example:

```
# cat /etc/nvme/hostnqn
nqn.2014-08.org.nvmexpress:uuid:325e7554-1f9b-11ec-8489-3a68dd61a4df


::> vserver nvme subsystem host show -vserver vs_nvme207
Vserver     Subsystem       Host NQN
----------- ---------------
------------------------------------------------------------
vs_nvme207 rhel_207_LPe32002     nqn.2014-
08.org.nvmexpress:uuid:325e7554-1f9b-11ec-8489-3a68dd61a4df
```

If the host NQN strings do not match, you should use the `vserver modify` command to update the host NQN string on your corresponding ONTAP NVMe subsystem to match the host NQN string `/etc/nvme/hostnqn` on the host.

5. Reboot the host.

Configure NVMe/FC

Broadcom/Emulex

Steps

1. Verify that you are using the supported adapter. For the most current list of supported adapters, see the NetApp Interoperability Matrix.

```
# cat /sys/class/scsi_host/host*/modelname
LPe32002-M2
LPe32002-M2

# cat /sys/class/scsi_host/host*/modeldesc

Emulex LightPulse LPe32002-M2 2-Port 32Gb Fibre Channel Adapter
Emulex LightPulse LPe32002-M2 2-Port 32Gb Fibre Channel Adapter
```

2. Verify that you are using the recommended Broadcom lpfc firmware and inbox driver. See the NetApp Interoperability Matrix for the most current list of supported adapter driver and firmware versions.

```
# cat /sys/class/scsi_host/host*/fwrev
14.0.505.11, sli-4:2:c
14.0.505.11, sli-4:2:c
```

```
# cat /sys/module/lpfc/version
0:14.2.0.5
```

3. Verify that `lpfc_enable_fc4_type` is set to 3

```
# cat /sys/module/lpfc/parameters/lpfc_enable_fc4_type
3
```

4. Verify that the initiator ports are up and running, and that you can see the target LIFs.

```
# cat /sys/class/fc_host/host*/port_name
0x100000109b1b95ef
0x100000109b1b95f0
```

```
# cat /sys/class/fc_host/host*/port_state
Online
Online
```

```
# cat /sys/class/scsi_host/host*/nvme_info
NVME Initiator Enabled
XRI Dist lpfc0 Total 6144 IO 5894 ELS 250
NVME LPORT lpfc0 WWPN x100000109b1b95ef WWNN x200000109b1b95ef DID
x061700 ONLINE
NVME RPORT       WWPN x2035d039ea1308e5 WWNN x2082d039ea1308e5 DID
x062f05 TARGET DISCSRVC ONLINE
NVME RPORT       WWPN x2083d039ea1308e5 WWNN x2082d039ea1308e5 DID
x062407 TARGET DISCSRVC ONLINE

NVME Statistics
LS: Xmt 000000000e Cmpl 000000000e Abort 00000000
LS XMIT: Err 00000000  CMPL: xb 00000000 Err 00000000
Total FCP Cmpl 000000000001df6c Issue 000000000001df6e OutIO
0000000000000002
        abort 00000000 noxri 00000000 nondlp 00000000 qdepth 00000000
wqerr 00000000 err 00000000
FCP CMPL: xb 00000000 Err 00000004

NVME Initiator Enabled
XRI Dist lpfc1 Total 6144 IO 5894 ELS 250
NVME LPORT lpfc1 WWPN x100000109b1b95f0 WWNN x200000109b1b95f0 DID
x061400 ONLINE
NVME RPORT       WWPN x2036d039ea1308e5 WWNN x2082d039ea1308e5 DID
x061605 TARGET DISCSRVC ONLINE
NVME RPORT       WWPN x2037d039ea1308e5 WWNN x2082d039ea1308e5 DID
x062007 TARGET DISCSRVC ONLINE

NVME Statistics
LS: Xmt 000000000e Cmpl 000000000e Abort 00000000
LS XMIT: Err 00000000  CMPL: xb 00000000 Err 00000000
Total FCP Cmpl 000000000001dd28 Issue 000000000001dd29 OutIO
0000000000000001
        abort 00000000 noxri 00000000 nondlp 00000000 qdepth 00000000
wqerr 00000000 err 00000000
FCP CMPL: xb 00000000 Err 00000004
```

Enable 1MB I/O size (Optional)

ONTAP reports an MDTS (Max Data Transfer Size) of 8 in the Identify Controller data which means the maximum I/O request size should be up to 1 MB. However, to issue I/O requests of size 1 MB for the Broadcom NVMe/FC host, the lpfc parameter lpfc_sg_seg_cnt should also be bumped up to 256 from the default value of 64. Use the following instructions to do so:

Steps

1. Append the value `256` in the respective `modprobe lpfc.conf` file:

```
# cat /etc/modprobe.d/lpfc.conf
options lpfc lpfc_sg_seg_cnt=256
```

2. Run a `dracut -f` command, and reboot the host.
3. After reboot, verify that the above setting has been applied by checking the corresponding `sysfs` value:

```
# cat /sys/module/lpfc/parameters/lpfc_sg_seg_cnt
256
```

Now the Broadcom FC-NVMe host should be able to send up to 1MB I/O requests on the ONTAP namespace devices.

Marvell/QLogic

The native inbox `qla2xxx` driver included in the RHEL 9.1 kernel has the latest upstream fixes which are essential for ONTAP support.

Steps

1. Verify that you are running the supported adapter driver and firmware versions using the following command:

```
# cat /sys/class/fc_host/host*/symbolic_name
QLE2772 FW:v9.08.02 DVR:v10.02.07.400-k-debug
QLE2772 FW:v9.08.02 DVR:v10.02.07.400-k-debug
```

2. Verify `ql2xnvmeenable` is set which enables the Marvell adapter to function as an NVMe/FC initiator using the following command:

```
# cat /sys/module/qla2xxx/parameters/ql2xnvmeenable
1
```

Configure NVMe/TCP

Unlike NVMe/FC, NVMe/TCP has no auto-connect functionality. This manifests two major limitations on the Linux NVMe/TCP host:

- **No auto-reconnect after paths get reinstated** NVMe/TCP cannot automatically reconnect to a path that is reinstated beyond the default `ctrl-loss-tmo` timer of 10 minutes following a path down.
- **No auto-connect during host boot** NVMe/TCP cannot connect automatically during host boot.

You should set the retry period for failover events to at least 30 minutes to prevent timeouts. You can increase the retry period by increasing the value of the `ctrl_loss_tmo timer` using the following procedure:

Steps

1. Verify whether the initiator port can fetch the discovery log page data across the supported NVMe/TCP LIFs:

```
# nvme discover -t tcp -w 192.168.1.8 -a 192.168.1.51

Discovery Log Number of Records 10, Generation counter 119
=====Discovery Log Entry 0======
trtype: tcp
adrfam: ipv4
subtype: nvme subsystem
treq: not specified
portid: 0
trsvcid: 4420
subnqn: nqn.1992-
08.com.netapp:sn.56e362e9bb4f11ebbaded039ea165abc:subsystem.nvme_118_tcp
_1
traddr: 192.168.2.56
sectype: none
=====Discovery Log Entry 1======
trtype: tcp
adrfam: ipv4
subtype: nvme subsystem
treq: not specified
portid: 1
trsvcid: 4420
subnqn: nqn.1992-
08.com.netapp:sn.56e362e9bb4f11ebbaded039ea165abc:subsystem.nvme_118_tcp
_1
traddr: 192.168.1.51
sectype: none
=====Discovery Log Entry 2======
trtype: tcp
adrfam: ipv4
subtype: nvme subsystem
treq: not specified
portid: 0
trsvcid: 4420
subnqn: nqn.1992-
08.com.netapp:sn.56e362e9bb4f11ebbaded039ea165abc:subsystem.nvme_118_tcp
_2
traddr: 192.168.2.56
sectype: none
...
```

2. Verify that the other NVMe/TCP initiator-target LIF combos can successfully fetch discovery log page data. For example:

```
# nvme discover -t tcp -w 192.168.1.8 -a 192.168.1.51
# nvme discover -t tcp -w 192.168.1.8 -a 192.168.1.52
# nvme discover -t tcp -w 192.168.2.9 -a 192.168.2.56
# nvme discover -t tcp -w 192.168.2.9 -a 192.168.2.57
```

3. Run `nvme connect-all` command across all the supported NVMe/TCP initiator-target LIFs across the nodes. Make sure you set a longer `ctrl_loss_tmo` timer retry period (for example, 30 minutes, which can be set through `-l 1800`) while running the `connect-all` command so that it would retry for a longer period of time in the event of a path loss. For example:

```
# nvme connect-all -t tcp -w 192.168.1.8 -a 192.168.1.51 -l 1800
# nvme connect-all -t tcp -w 192.168.1.8 -a 192.168.1.52 -l 1800
# nvme connect-all -t tcp -w 192.168.2.9 -a 192.168.2.56 -l 1800
# nvme connect-all -t tcp -w 192.168.2.9 -a 192.168.2.57 -l 1800
```

Validate NVMe-oF

Steps

1. Verify that in-kernel NVMe multipath is indeed enabled by checking:

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

2. Verify that the appropriate NVMe-oF settings (such as, `model` set to `NetApp ONTAP Controller` and load balancing `iopolicy` set to `round-robin`) for the respective ONTAP namespaces properly reflect on the host:

```
# cat /sys/class/nvme-subsystem/nvme-subsys*/model
NetApp ONTAP Controller
NetApp ONTAP Controller
```

```
# cat /sys/class/nvme-subsystem/nvme-subsys*/iopolicy
round-robin
round-robin
```

3. Verify that the ONTAP namespaces properly reflect on the host. For example:

```
# nvme list
Node            SN                    Model                     Namespace
------------    --------------------- ---------------------------------
/dev/nvme0n1    81CZ5BQuUNfGAAAAAAAB   NetApp ONTAP Controller   1

Usage                 Format         FW Rev
--------------------  -----------    --------
85.90 GB / 85.90 GB   4 KiB + 0 B    FFFFFFFF
```

4. Verify that the controller state of each path is live and has proper ANA status. For example:

   Example (a):

```
# nvme list-subsys /dev/nvme0n1
nvme-subsys10 - NQN=nqn.1992-
08.com.netapp:sn.82e7f9edc72311ec8187d039ea14107d:subsystem.rhel_131_QLe
2742
\
 +- nvme2 fc traddr=nn-0x2038d039ea1308e5:pn-
0x2039d039ea1308e5,host_traddr=nn-0x20000024ff171d30:pn-
0x21000024ff171d30 live non-optimized
 +- nvme3 fc traddr=nn-0x2038d039ea1308e5:pn-
0x203cd039ea1308e5,host_traddr=nn-0x20000024ff171d31:pn-
0x21000024ff171d31 live optimized
 +- nvme4 fc traddr=nn-0x2038d039ea1308e5:pn-
0x203bd039ea1308e5,host_traddr=nn-0x20000024ff171d30:pn-
0x21000024ff171d30 live optimized
 +- nvme5 fc traddr=nn-0x2038d039ea1308e5:pn-
0x203ad039ea1308e5,host_traddr=nn-0x20000024ff171d31:pn-
0x21000024ff171d31 live non-optimized
```

   Example (b):

```
# nvme list-subsys /dev/nvme0n1
nvme-subsys1 - NQN=nqn.1992-
08.com.netapp:sn.bf0691a7c74411ec8187d039ea14107d:subsystem.rhel_tcp_133
\
 +- nvme1 tcp
traddr=192.168.166.21,trsvcid=4420,host_traddr=192.168.166.5 live non-
optimized
 +- nvme2 tcp
traddr=192.168.166.20,trsvcid=4420,host_traddr=192.168.166.5 live
optimized
 +- nvme3 tcp
traddr=192.168.167.21,trsvcid=4420,host_traddr=192.168.167.5 live non-
optimized
 +- nvme4 tcp
traddr=192.168.167.20,trsvcid=4420,host_traddr=192.168.167.5 live
optimized
```

5. Verify that the NetApp plug-in displays proper values for each ONTAP namespace device. For example:

```
# nvme netapp ontapdevices -o column
Device       Vserver          Namespace Path
---------    -------
-------------------------------------------------
/dev/nvme0n1 vs_tcp79     /vol/vol1/ns1

NSID  UUID                                   Size
----  ------------------------------         ------
1     79c2c569-b7fa-42d5-b870-d9d6d7e5fa84  21.47GB


# nvme netapp ontapdevices -o json
{

  "ONTAPdevices" : [
  {

      "Device" : "/dev/nvme0n1",
      "Vserver" : "vs_tcp79",
      "Namespace_Path" : "/vol/vol1/ns1",
      "NSID" : 1,
      "UUID" : "79c2c569-b7fa-42d5-b870-d9d6d7e5fa84",
      "Size" : "21.47GB",
      "LBA_Data_Size" : 4096,
      "Namespace_Size" : 5242880
    },

]

}
```

Example (b)

```
# nvme netapp ontapdevices -o column

Device           Vserver                   Namespace Path
---------------- ------------------------
----------------------------------
/dev/nvme1n1     vs_tcp_133                /vol/vol1/ns1

NSID UUID                                   Size
----------------------------------------------------------
1    1ef7cb56-bfed-43c1-97c1-ef22eeb92657   21.47GB

# nvme netapp ontapdevices -o json
{
  "ONTAPdevices":[
    {
      "Device":"/dev/nvme1n1",
      "Vserver":"vs_tcp_133",
      "Namespace_Path":"/vol/vol1/ns1",
      "NSID":1,
      "UUID":"1ef7cb56-bfed-43c1-97c1-ef22eeb92657",
      "Size":"21.47GB",
      "LBA_Data_Size":4096,
      "Namespace_Size":5242880
    },
  ]

}
```

Troubleshooting

Before commencing any troubleshooting for any NVMe/FC failures, make sure that you are running a configuration that is compliant to the Interoperability Matrix Tool (IMT) specifications and then proceed with the next steps to debug any host side issues.

LPFC verbose logging

Steps

1. Set the `lpfc_log_verbose` driver setting to any of the following values to log NVMe/FC events:

```
#define LOG_NVME 0x00100000 /* NVME general events. */
#define LOG_NVME_DISC 0x00200000 /* NVME Discovery/Connect events. */
#define LOG_NVME_ABTS 0x00400000 /* NVME ABTS events. */
#define LOG_NVME_IOERR 0x00800000 /* NVME IO Error events. */
```

2. After setting any of these values, run `dracut-f` command to recreate the `initramfs` and reboot the host.
3. After rebooting, verify the settings:

```
# cat /etc/modprobe.d/lpfc.conf
options lpfc lpfc_log_verbose=0xf00083


# cat /sys/module/lpfc/parameters/lpfc_log_verbose
15728771
```

qla2xxx verbose logging

There is no similar specific qla2xxx logging for NVMe/FC as for the `lpfc` driver. Therefore, you can set the general qla2xxx logging level using the following steps:

Steps

1. Append the `ql2xextended_error_logging=0x1e400000` value to the corresponding `modprobe qla2xxx conf` file.
2. Recreate the `initramfs` by running `dracut -f` command and then reboot the host.
3. After reboot, verify that the verbose logging has been applied as follows:

```
# cat /etc/modprobe.d/qla2xxx.conf
options qla2xxx ql2xnvmeenable=1 ql2xextended_error_logging=0x1e400000
# cat /sys/module/qla2xxx/parameters/ql2xextended_error_logging
507510784
```

Known issues

NetApp Bug ID	Title	Description	Bugzilla ID
1503468	`nvme list-subsys` command returns repeated nvme controller list for a given subsystem	The `nvme list-subsys` command should return a unique list of nvme controllers associated to a given subsystem. In RHEL 9.1, the `nvme list-subsys` command returns nvme controllers with its respective ANA state for all namespaces that belong to a given subsystem. However, the ANA state is a per-namespace attribute therefore, it would be ideal to display unique nvme controller entries with the path state if you list the subsystem command syntax for a given namespace.	2130106

Common nvme-cli errors and workarounds

The errors displayed by `nvme-cli` during `nvme discover`, `nvme connect` or `nvme connect-all` operations and the workarounds are shown in the following table:

Errors displayed by `nvme-cli`	**Probable cause**	**Workaround**
`Failed to write to /dev/nvme-fabrics: Invalid argument`	Incorrect syntax	Ensure you are using the correct syntax for the above nvme commands.

Errors displayed by `nvme-cli`	Probable cause	Workaround

`Failed to write to /dev/nvme-fabrics: No such file or directory`	Multiple issues could trigger this.
	Passing wrong arguments to the nvme commands is one of the common causes.
		• Ensure you have passed the correct arguments (such as, correct WWNN string, WWPN string, and more) to the commands.
		• If the arguments are correct, but you still see this error, check if the `/sys/class/scsi_host/host*/nvme_info` output is proper, the NVMe initiator showing as `Enabled`, and the NVMe/FC target LIFs properly showing up here under the remote ports sections.
		Example:
		`# cat /sys/class/scsi_host/host*/nvme_info`
		`NVME Initiator Enabled`
		`NVME LPORT lpfc0 WWPN x10000090fae0ec9d WWNN x20000090fae0ec9d DID x012000 ONLINE`
		`NVME RPORT WWPN x200b00a098c80f09 WWNN x200a00a098c80f09 DID x010601 TARGET DISCSRVC ONLINE`
		`NVME Statistics`
		`LS: Xmt 0000000000000006 Cmpl 0000000000000006`
		`FCP: Rd 0000000000000071 Wr 0000000000000005 IO 0000000000000031 Cmpl 00000000000000a6 Outstanding 0000000000000001`
		`NVME Initiator Enabled`
		`NVME LPORT lpfc1 WWPN x10000090fae0ec9e WWNN x20000090fae0ec9e DID x012400 ONLINE`
		`NVME RPORT WWPN x200900a098c80f09 WWNN x200800a098c80f09 DID x010301 TARGET DISCSRVC ONLINE`
		`NVME Statistics`
		`LS: Xmt 0000000000000006 Cmpl 0000000000000006`
		`FCP: Rd 0000000000000073 Wr 0000000000000005 IO 0000000000000031 Cmpl 00000000000000a8 Outstanding 0000000000000001`
		• If the target LIFs don't show up as above in the nvme_info output, check the `/var/log/messages` and `dmesg` output for any suspicious NVMe/FC failures, and report or fix accordingly.

Errors displayed by `nvme-cli`	Probable cause	Workaround
`No discovery log entries to fetch`	Generally seen if the `/etc/nvme/hostnqn` string has not been added to the corresponding subsystem on the NetApp array or an incorrect `hostnqn` string has been added to the respective subsystem.	Ensure the exact `/etc/nvme/hostnqn` string is added to the corresponding subsystem on the NetApp array (verify through the `vserver nvme subsystem host show` command).
`Failed to write to /dev/nvme-fabrics: Operation already in progress`	Seen if the controller associations or specified operation is already created or in the process of being created. This could happen as part of the auto-connect scripts installed above.	None. For `nvme discover`, try running this command after some time. For `nvme connect` and `connect-all`, run the `nvme list` command to verify that the namespace devices are already created and displayed on the host.

When to contact technical support

If you are still facing issues, collect the following files and command outputs and contact technical support for further triage:

```
cat /sys/class/scsi_host/host*/nvme_info
/var/log/messages
dmesg
nvme discover output as in:
nvme discover --transport=fc --traddr=nn-0x200a00a098c80f09:pn
-0x200b00a098c80f09 --host-traddr=nn-0x20000090fae0ec9d:pn
-0x10000090fae0ec9d
nvme list
nvme list-subsys /dev/nvmeXnY
```

NVMe-oF Host Configuration for RHEL 9.0 with ONTAP

Supportability

NVMe-oF (including NVMe/FC and NVMe/TCP) is supported with RHEL 9.0 with Asymmetric Namespace Access (ANA) required for surviving storage failovers (SFOs) on the ONTAP array. ANA is the ALUA equivalent in the NVM-oF environment, and is currently implemented with in-kernel NVMe Multipath. This document contains the details for enabling NVMe-oF with in-kernel NVMe Multipath using ANA on RHEL 9.0 and ONTAP as the target.

You can use the configuration settings provided in this document to configure cloud clients connected to Cloud Volumes ONTAP and Amazon FSx for ONTAP.

Features

- Starting RHEL 9.0, NVMe/TCP is no longer a technology preview feature (unlike RHEL 8) but a fully supported enterprise feature itself.
- Starting RHEL 9.0, in-kernel NVMe multipath is enabled for NVMe namespaces by default, without the need for explicit settings (unlike RHEL 8).

Configuration requirements

Refer to the NetApp Interoperability Matrix for exact details regarding supported configurations.

Enable in-kernel NVMe Multipath

Steps

1. Install RHEL 9.0 on the server. After the installation is complete, verify that you are running the specified RHEL 9.0 kernel. See NetApp Interoperability Matrix for the most current list of supported versions.
2. After the installation is complete, verify that you are running the specified RHEL 9.0 kernel. See NetApp Interoperability Matrix for the most current list of supported versions.

```
# uname -r
5.14.0-70.13.1.el9_0.x86_64
```

3. Install the `nvme-cli` package.

```
# rpm -qa|grep nvme-cli
nvme-cli-1.16-3.el9.x86_64
```

4. On the host, check the host NQN string at `/etc/nvme/hostnqn` and verify that it matches the host NQN string for the corresponding subsystem on the ONTAP array. For example,

```
# cat /etc/nvme/hostnqn
nqn.2014-08.org.nvmexpress:uuid:9ed5b327-b9fc-4cf5-97b3-1b5d986345d1
```

```
::> vserver nvme subsystem host show -vserver vs_fcnvme_141
Vserver     Subsystem Host     NQN
----------- ---------------
------------------------------------------------------------
vs_fcnvme_14 nvme_141_1 nqn.2014-08.org.nvmexpress:uuid:9ed5b327-b9fc-
4cf5-97b3-1b5d986345d1
```

If the host NQN strings do not match, you should use the `vserver modify` command to update the host NQN string on your corresponding ONTAP NVMe subsystem to match the host NQN string from `/etc/nvme/hostnqn` on the host.

5. Reboot the host.

Configure NVMe/FC

Broadcom/Emulex

1. Verify that you are using the supported adapter. For the most current list of supported adapters see NetApp Interoperability Matrix.

```
# cat /sys/class/scsi_host/host*/modelname
LPe32002-M2
LPe32002-M2
```

```
# cat /sys/class/scsi_host/host*/modeldesc
Emulex LightPulse LPe32002-M2 2-Port 32Gb Fibre Channel Adapter
Emulex LightPulse LPe32002-M2 2-Port 32Gb Fibre Channel Adapter
```

2. Verify that you are using the recommended Broadcom lpfc firmware and inbox driver. For the most current list of supported adapter driver and firmware versions, see NetApp Interoperability Matrix.

```
# cat /sys/class/scsi_host/host*/fwrev
12.8.351.47, sli-4:2:c
12.8.351.47, sli-4:2:c
```

```
# cat /sys/module/lpfc/version
0:14.0.0.4
```

3. Verify that `lpfc_enable_fc4_type` is set to `3`.

```
# cat /sys/module/lpfc/parameters/lpfc_enable_fc4_type
3
```

4. Verify that the initiator ports are up and running, and you are able to see the target LIFs.

```
# cat /sys/class/fc_host/host*/port_name
0x100000109b1c1204
0x100000109b1c1205
```

```
# cat /sys/class/fc_host/host*/port_state
Online
Online
```

```
# cat /sys/class/scsi_host/host*/nvme_info

NVME Initiator Enabled
XRI Dist lpfc0 Total 6144 IO 5894 ELS 250
NVME LPORT lpfc0 WWPN x100000109b1c1204 WWNN x200000109b1c1204 DID
x011d00 ONLINE
NVME RPORT WWPN x203800a098dfdd91 WWNN x203700a098dfdd91 DID x010c07
TARGET DISCSRVC ONLINE
NVME RPORT WWPN x203900a098dfdd91 WWNN x203700a098dfdd91 DID x011507
TARGET DISCSRVC ONLINE

NVME Statistics
LS: Xmt 0000000f78 Cmpl 0000000f78 Abort 00000000
LS XMIT: Err 00000000 CMPL: xb 00000000 Err 00000000
Total FCP Cmpl 000000002fe29bba Issue 000000002fe29bc4 OutIO
000000000000000a
abort 00001bc7 noxri 00000000 nondlp 00000000 qdepth 00000000 wqerr
00000000 err 00000000
FCP CMPL: xb 00001e15 Err 0000d906

NVME Initiator Enabled
XRI Dist lpfc1 Total 6144 IO 5894 ELS 250
NVME LPORT lpfc1 WWPN x100000109b1c1205 WWNN x200000109b1c1205 DID
x011900 ONLINE
NVME RPORT WWPN x203d00a098dfdd91 WWNN x203700a098dfdd91 DID x010007
TARGET DISCSRVC ONLINE
NVME RPORT WWPN x203a00a098dfdd91 WWNN x203700a098dfdd91 DID x012a07
TARGET DISCSRVC ONLINE

NVME Statistics
LS: Xmt 0000000fa8 Cmpl 0000000fa8 Abort 00000000
LS XMIT: Err 00000000 CMPL: xb 00000000 Err 00000000
Total FCP Cmpl 000000002e14f170 Issue 000000002e14f17a OutIO
000000000000000a
abort 000016bb noxri 00000000 nondlp 00000000 qdepth 00000000 wqerr
00000000 err 00000000
FCP CMPL: xb 00001f50 Err 0000d9f8
```

5. Enable 1MB I/O size.

The `lpfc_sg_seg_cnt` parameter needs to be set to `256` for the `lpfc` driver to issue I/O requests upto 1 MB size.

```
# cat /etc/modprobe.d/lpfc.conf
options lpfc lpfc_sg_seg_cnt=256
```

a. Run a `dracut -f` command and then reboot the host.

b. After the host boots up, verify that `lpfc_sg_seg_cnt` is set to `256`.

```
# cat /sys/module/lpfc/parameters/lpfc_sg_seg_cnt
256
```

Marvell/QLogic

The native inbox qla2xxx driver included in the RHEL 9.0 kernel has the latest upstream fixes, essential for ONTAP support. Verify that you are running the supported adapter driver and firmware versions:

```
# cat /sys/class/fc_host/host*/symbolic_name
QLE2742 FW:v9.06.02 DVR:v10.02.00.200-k
QLE2742 FW:v9.06.02 DVR:v10.02.00.200-k
```

Verify `ql2xnvmeenable` is set which enables the Marvell adapter to function as a NVMe/FC initiator:

```
# cat /sys/module/qla2xxx/parameters/ql2xnvmeenable
1
```

Configure NVMe/TCP

Unlike NVMe/FC, NVMe/TCP has no auto-connect functionality. This manifests two major limitations on the Linux NVMe/TCP host:

- **No auto-reconnect after paths get reinstated** NVMe/TCP cannot automatically reconnect to a path that is reinstated beyond the default `ctrl-loss-tmo` timer of 10 minutes following a path down.
- **No auto-connect during host bootup** NVMe/TCP cannot automatically connect during host bootup as well.

You should set the retry period for failover events to at least 30 minutes to prevent timeouts. You can increase the retry period by increasing the value of the ctrl_loss_tmo timer. Following are the details:

Steps

1. Verify whether the initiator port is able to fetch discovery log page data across the supported NVMe/TCP LIFs:

```
# nvme discover -t tcp -w 192.168.1.8 -a 192.168.1.51

Discovery Log Number of Records 10, Generation counter 119
=====Discovery Log Entry 0======
trtype: tcp
adrfam: ipv4
subtype: nvme subsystem
treq: not specified
portid: 0
trsvcid: 4420
subnqn: nqn.1992-
08.com.netapp:sn.56e362e9bb4f11ebbaded039ea165abc:subsystem.nvme_118_tcp
_1
traddr: 192.168.2.56
sectype: none
=====Discovery Log Entry 1======
trtype: tcp
adrfam: ipv4
subtype: nvme subsystem
treq: not specified
portid: 1
trsvcid: 4420
subnqn: nqn.1992-
08.com.netapp:sn.56e362e9bb4f11ebbaded039ea165abc:subsystem.nvme_118_tcp
_1
traddr: 192.168.1.51
sectype: none
=====Discovery Log Entry 2======
trtype: tcp
adrfam: ipv4
subtype: nvme subsystem
treq: not specified
portid: 0
trsvcid: 4420
subnqn: nqn.1992-
08.com.netapp:sn.56e362e9bb4f11ebbaded039ea165abc:subsystem.nvme_118_tcp
_2
traddr: 192.168.2.56
sectype: none
...
```

2. Similarly, verify that the other NVMe/TCP initiator-target LIF combos are able to successfully fetch the discovery log page data. For example,

```
# nvme discover -t tcp -w 192.168.1.8 -a 192.168.1.51
# nvme discover -t tcp -w 192.168.1.8 -a 192.168.1.52
# nvme discover -t tcp -w 192.168.2.9 -a 192.168.2.56
# nvme discover -t tcp -w 192.168.2.9 -a 192.168.2.57
```

3. Run `nvme connect-all` command across all the supported NVMe/TCP initiator-target LIFs across the nodes. Ensure you set a longer `ctrl_loss_tmo` timer retry period (for example, 30 minutes, which can be set through `-l 1800`) during the connect-all so that it would retry for a longer period of time in the event of a path loss. For example,

```
# nvme connect-all -t tcp -w 192.168.1.8 -a 192.168.1.51 -l 1800
# nvme connect-all -t tcp -w 192.168.1.8 -a 192.168.1.52 -l 1800
# nvme connect-all -t tcp -w 192.168.2.9 -a 192.168.2.56 -l 1800
# nvme connect-all -t tcp -w 192.168.2.9 -a 192.168.2.57 -l 1800
```

Validate NVMf

Steps

1. Verify that in-kernel NVMe multipath is indeed enabled by checking:

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

2. Verify that the appropriate NVMf settings (for example, model set to `NetApp ONTAP Controller` and load balancing `iopolicy` set to `round-robin`) for the respective ONTAP namespaces properly reflect on the host:

```
# cat /sys/class/nvme-subsystem/nvme-subsys*/model
NetApp ONTAP Controller
NetApp ONTAP Controller
```

```
# cat /sys/class/nvme-subsystem/nvme-subsys*/iopolicy
round-robin
round-robin
```

3. Verify that the ONTAP namespaces properly reflect on the host. For example (a),

```
# nvme list
Node         SN                    Model                    Namespace
Usage
------      ---------------------------------------
------------------------
/dev/nvme0n1 814vWBNRwf9HAAAAAAAB  NetApp ONTAP Controller  1
85.90 GB / 85.90 GB

Format         FW Rev
---------------------
4 KiB + 0 B   FFFFFFFF
```

Example (b):

```
# nvme list
Node           SN                   Model                     Namespace
Usage
-----------------------------------------------------
-------------------------------------
/dev/nvme0n1   81CZ5BQuUNfGAAAAAAAB NetApp ONTAP Controller   1
85.90 GB / 85.90 GB

Format         FW Rev
-----------------------
4 KiB + 0 B   FFFFFFFF
```

4. Verify that the controller state of each path is live and has a proper ANA status.
For example (a),

```
# nvme list-subsys /dev/nvme0n1
nvme-subsys0 - NQN=nqn.1992-
08.com.netapp:sn.5f5f2c4aa73b11e9967e00a098df41bd:subsystem.nvme_141_1
\
+- nvme0 fc traddr=nn-0x203700a098dfdd91:pn-0x203800a098dfdd91
host_traddr=nn-0x200000109b1c1204:pn-0x100000109b1c1204 live
inaccessible
+- nvme1 fc traddr=nn-0x203700a098dfdd91:pn-0x203900a098dfdd91
host_traddr=nn-0x200000109b1c1204:pn-0x100000109b1c1204 live
inaccessible
+- nvme2 fc traddr=nn-0x203700a098dfdd91:pn-0x203a00a098dfdd91
host_traddr=nn-0x200000109b1c1205:pn-0x100000109b1c1205 live optimized
+- nvme3 fc traddr=nn-0x203700a098dfdd91:pn-0x203d00a098dfdd91
host_traddr=nn-0x200000109b1c1205:pn-0x100000109b1c1205 live optimized
```

Example (b):

```
# nvme list-subsys /dev/nvme0n1
nvme-subsys0 - NQN=nqn.1992-
08.com.netapp:sn.56e362e9bb4f11ebbaded039ea165abc:subsystem.nvme_118_tcp
_1
\
+- nvme0 tcp traddr=192.168.1.51 trsvcid=4420 host_traddr=192.168.1.8
live optimized
+- nvme10 tcp traddr=192.168.2.56 trsvcid=4420 host_traddr=192.168.2.9
live optimized
+- nvme15 tcp traddr=192.168.2.57 trsvcid=4420 host_traddr=192.168.2.9
live non-optimized
+- nvme5 tcp traddr=192.168.1.52 trsvcid=4420 host_traddr=192.168.1.8
live non-optimized
```

5. Verify the NetApp plug-in displays proper values for each ONTAP namespace device.
For example (a),

```
# nvme netapp ontapdevices -o column
Device       Vserver        Namespace Path
NSID
---------------------- ----------------------------
------------------------
/dev/nvme0n1  vs_fcnvme_141  /vol/fcnvme_141_vol_1_1_0/fcnvme_141_ns   1

UUID                                   Size
----------------------------------------
72b887b1-5fb6-47b8-be0b-33326e2542e2   85.90GB

# nvme netapp ontapdevices -o json
{
"ONTAPdevices" : [
    {
        "Device" : "/dev/nvme0n1",
        "Vserver" : "vs_fcnvme_141",
        "Namespace_Path" : "/vol/fcnvme_141_vol_1_1_0/fcnvme_141_ns",
        "NSID" : 1,
        "UUID" : "72b887b1-5fb6-47b8-be0b-33326e2542e2",
        "Size" : "85.90GB",
        "LBA_Data_Size" : 4096,
        "Namespace_Size" : 20971520
    }
  ]
}
```

Example (b):

```
# nvme netapp ontapdevices -o column
Device               Vserver                   Namespace Path
---------------------- -------------------------
-----------------------------------
/dev/nvme0n1         vs_tcp_118
/vol/tcpnvme_118_1_0_0/tcpnvme_118_ns

NSID   UUID                                   Size
-------------------------------------------------
1     4a3e89de-b239-45d8-be0c-b81f6418283c 85.90GB
```

```
# nvme netapp ontapdevices -o json
{
"ONTAPdevices" : [
    {
     "Device" : "/dev/nvme0n1",
      "Vserver" : "vs_tcp_118",
      "Namespace_Path" : "/vol/tcpnvme_118_1_0_0/tcpnvme_118_ns",
      "NSID" : 1,
      "UUID" : "4a3e89de-b239-45d8-be0c-b81f6418283c",
      "Size" : "85.90GB",
      "LBA_Data_Size" : 4096,
      "Namespace_Size" : 20971520
    },
  ]

}
```

Troubleshooting

Before commencing any troubleshooting for any NVMe/FC failures, always ensure you are running a configuration that is compliant to the IMT specifications. And then proceed to the following steps to debug any host side issues.

lpfc verbose logging

Following is the list of lpfc driver logging bitmasks available for NVMe/FC, as seen at `drivers/scsi/lpfc/lpfc_logmsg.h`:

```
#define LOG_NVME 0x00100000 /* NVME general events. */
#define LOG_NVME_DISC 0x00200000 /* NVME Discovery/Connect events. */
#define LOG_NVME_ABTS 0x00400000 /* NVME ABTS events. */
#define LOG_NVME_IOERR 0x00800000 /* NVME IO Error events. */
```

You can set the `lpfc_log_verbose` driver setting (appended to the lpfc line at `/etc/modprobe.d/lpfc.conf`) to any of the values above for logging NVMe/FC events from a `lpfc` driver perspective. And then recreate the `initiramfs` by running `dracut -f` command and then reboot the host. After rebooting, verify that the verbose logging has applied by checking the following, using the above `LOG_NVME_DISC` bitmask as an example:

```
# cat /etc/modprobe.d/lpfc.conf
options lpfc_enable_fc4_type=3 lpfc_log_verbose=0xf00083
```

```
# cat /sys/module/lpfc/parameters/lpfc_log_verbose
15728771
```

qla2xxx verbose logging

There is no similar specific qla2xxx logging for NVMe/FC, as is there in `lpfc`. You can set the general qla2xxx logging level here, for example, `ql2xextended_error_logging=0x1e400000`. This can be done by appending this value to the corresponding `modprobe qla2xxx conf` file. And then recreate the `initramfs` by running `dracut -f` and then reboot the host. After reboot, verify that the verbose logging has applied as follows:

```
# cat /etc/modprobe.d/qla2xxx.conf
options qla2xxx ql2xnvmeenable=1 ql2xextended_error_logging=0x1e400000
```

```
# cat /sys/module/qla2xxx/parameters/ql2xextended_error_logging
507510784
```

Known issues

NetApp Bug ID	Title	Description	Bugzilla ID
1479047	RHEL 9.0 NVMe-oF hosts create duplicate Persistent Discovery Controllers	On NVMe over Fabrics (NVMe-oF) hosts, you can use the "nvme discover -p" command to create Persistent Discovery Controllers (PDCs). When this command is used, only one PDC should be created per initiator-target combination. However, if you are running ONTAP 9.10.1 and Red Hat Enterprise Linux (RHEL) 9.0 with an NVMe-oF host, a duplicate PDC is created each time "nvme discover -p" is executed. This leads to unnecessary usage of resources on both the host and the target.	2087000

Common nvme-cli errors and workarounds

Errors displayed by `nvme-cli`	Probable cause	Workaround
`Failed to write to /dev/nvme-fabrics: Invalid argument` error during `nvme discover`, `nvme connect`, or `nvme connect-all`	This error message is generally displayed if the syntax is wrong.	Ensure you are using the correct syntax for the above nvme commands.

Errors displayed by `nvme-cli`	**Probable cause**	**Workaround**

`Failed to write to /dev/nvme-fabrics: No such file or directory` during `nvme discover`, `nvme connect`, or `nvme connect-all`	Multiple issues could trigger this. Some of the common cases are:
	
	You passed wrong arguments to the above nvme commands.
		Ensure you have passed the appropriate arguments (such as appropriate WWNN string, WWPN string, and more) for the above commands.
		If the arguments are correct, but still seeing this error, check if the `/sys/class/scsi_host/host*/nvme_info` output is proper with the NVMe initiator showing as `Enabled` and NVMe/FC target LIFs properly showing up here under the remote ports sections. For example,
		
		`# cat /sys/class/scsi_host/host*/nvme_info`
		`NVME Initiator Enabled`
		`NVME LPORT lpfc0 WWPN x10000090fae0ec9d WWNN x20000090fae0ec9d DID x012000 ONLINE`
		`NVME RPORT WWPN x200b00a098c80f09 WWNN x200a00a098c80f09 DID x010601 TARGET DISCSRVC ONLINE`
		
		`NVME Statistics`
		`LS: Xmt 0000000000000006 Cmpl 0000000000000006`
		`FCP: Rd 0000000000000071 Wr 0000000000000005 IO 0000000000000031 Cmpl 00000000000000a6 Outstanding 0000000000000001`
		
		`NVME Initiator Enabled`
		`NVME LPORT lpfc1 WWPN x10000090fae0ec9e WWNN x20000090fae0ec9e DID x012400 ONLINE`
		`NVME RPORT WWPN x200900a098c80f09 WWNN x200800a098c80f09 DID x010301 TARGET DISCSRVC ONLINE`
		
		`NVME Statistics`
		`LS: Xmt 0000000000000006 Cmpl 0000000000000006`
		`FCP: Rd 0000000000000073 Wr 0000000000000005 IO 0000000000000031 Cmpl 00000000000000a8 Outstanding 0000000000000001`
		
		Workaround: If the target LIFs don't show up as above in the `nvme_info` output, check the `/var/log/messages` and `dmesg` output for any suspicious NVMe/FC failures, and report or fix accordingly.

Errors displayed by `nvme-cli`	Probable cause	Workaround
`No discovery log entries to fetch` during `nvme discover`, `nvme connect`, or `nvme connect-all`	This error message is generally seen if the `/etc/nvme/hostnqn` string has not been added to the corresponding subsystem on the NetApp array or an incorrect `hostnqn` string has been added to the respective subsystem.	Ensure the exact `/etc/nvme/hostnqn` string is added to the corresponding subsystem on the NetApp array (verify through the `vserver nvme subsystem host show`).
`Failed to write to /dev/nvme-fabrics: Operation already in progress` during `nvme discover`, `nvme connect` or `nvme connect-all`	This error message is seen if the controller associations or specified operation is already created or in the process of being created. This could happen as part of the auto-connect scripts installed above.	None. For `nvme discover`, try running this command after some time. And for `nvme connect` and `connect-all`, run a `nvme list` to verify that the namespace devices are already created and displayed on the host.

When to contact technical support

If you are still facing issues, please collect the following files and command outputs and send them for further triage:

```
cat /sys/class/scsi_host/host*/nvme_info
/var/log/messages
dmesg
nvme discover output as in:
nvme discover --transport=fc --traddr=nn-0x200a00a098c80f09:pn
-0x200b00a098c80f09 --host-traddr=nn-0x20000090fae0ec9d:pn
-0x10000090fae0ec9d
nvme list
nvme list-subsys /dev/nvmeXnY
```

Copyright information

Trademark information