RHEL

ONTAP SAN Host Utilities

NetApp
January 29, 2026

This PDF was generated from https://docs.netapp.com/zh-tw/ontap-sanhost/nvme-rhel-supported-features.html on January 29, 2026. Always check docs.netapp.com for the latest.

目錄

RHEL . . . 1
了解ONTAP對 RHEL 主機的支援和功能 . . . 1
下一步 . . . 1
配置 RHEL 10.x 以使用 NVMe-oF 和ONTAP存儲 . . . 2
步驟 1 ：選擇性啟用 SAN 開機 . . . 2
步驟 2：安裝 RHEL 和 NVMe 軟體並驗證您的配置 . . . 2
步驟 3：設定 NVMe/FC 和 NVMe/TCP . . . 4
步驟 4：（可選）修改 udev 規則中的 iopolicy . . . 12
步驟 5：可選，啟用 NVMe/FC 的 1MB I/O。 . . . 13
步驟 6：驗證 NVMe 啟動服務 . . . 13
步驟 7：驗證多路徑配置 . . . 15
步驟 8：設定安全帶內身份驗證 . . . 20
步驟 9 ：檢閱已知問題 . . . 26
配置 RHEL 9.x 以使用 NVMe-oF 和ONTAP存儲 . . . 26
步驟 1 ：選擇性啟用 SAN 開機 . . . 26
步驟 2：安裝 RHEL 和 NVMe 軟體並驗證您的配置 . . . 26
步驟 3：設定 NVMe/FC 和 NVMe/TCP . . . 28
步驟 4：（可選）修改 udev 規則中的 iopolicy . . . 37
步驟 5：可選，啟用 NVMe/FC 的 1MB I/O。 . . . 38
步驟 6：驗證 NVMe 啟動服務 . . . 39
步驟 7：驗證多路徑配置 . . . 40
步驟 8：設定安全帶內身份驗證 . . . 45
步驟 9 ：檢閱已知問題 . . . 51
配置 RHEL 8.x 以使用 NVMe-oF 和ONTAP存儲 . . . 52
步驟 1 ：選擇性啟用 SAN 開機 . . . 52
步驟 2：安裝 RHEL 和 NVMe 軟體並驗證您的配置 . . . 52
步驟 3：設定 NVMe/FC 和 NVMe/TCP . . . 54
步驟 4：（可選）為 NVMe/FC 啟用 1MB I/O . . . 60
步驟 5：驗證多路徑配置 . . . 60
步驟 6 ：檢閱已知問題 . . . 65

RHEL

了解ONTAP對 RHEL 主機的支援和功能

使用 NVMe over Fabrics (NVMe-oF) 進行主機配置所支援的功能會因ONTAP和 RHEL 的版本而異。

特徵	**RHEL** 主機版本	**ONTAP** 版本
RHEL 主機與ONTAP控制器之間透過 NVMe/TCP 支援安全的帶內驗證。	9.3 或更高版本	9.12.1 或更高版本
NVMe/TCP 使用原生命名空間提供命名空間 `nvme-cli` 包裹	8.2 或更高版本	9.10.1 或更高版本
NVMe/TCP 是一項完全支援的企業級功能	9.0 或更高版本	9.10.1 或更高版本
在同一主機上支援 NVMe 和 SCSI 流量，NVMe-oF 命名空間使用 NVMe 多路徑，SCSI LUN 使用 dm 多路徑。	8.2 或更高版本	9.4 或更高版本

無論系統設定中執行的ONTAP版本為何， ONTAP支援以下 SAN 主機功能。

特徵	**RHEL** 主機版本
預設啟用原生 NVMe 多路徑。	10.0 或更高版本
`nvme-cli` 軟體套件中的原生 udev 規則為 NVMe 多路徑提供佇列深度負載平衡	9.6 或更高版本
SAN 啟動已透過 NVMe/FC 協定啟用	9.4 或更高版本
這 `nvme-cli`軟體包包含自動連接腳本，無需第三方腳本。	8.2 或更高版本
`nvme-cli` 軟體套件中的原生 udev 規則為 NVMe 多路徑提供輪詢負載平衡	8.2 或更高版本

有關受支援配置的詳細信息，請參閱"互通性對照表工具"。

下一步

如果您的 **RHEL** 版本是...	了解更多...
10系列	"為 RHEL 10.x 設定 NVMe"
9系列	"為 RHEL 9.x 設定 NVMe"
8系列	"為 RHEL 8.x 設定 NVMe"

相關資訊

"了解如何管理 NVMe 協議"

配置 RHEL 10.x 以使用 NVMe-oF 和ONTAP存儲

Red Hat Enterpirse Linux (RHEL) 主機支援具有非對稱命名空間存取 (ANA) 的 NVMe over Fibre Channel (NVMe/FC) 和 NVMe over TCP (NVMe/TCP) 協定。 ANA 提供與 iSCSI 和 FCP 環境中的非對稱邏輯單元存取 (ALUA) 等效的多路徑功能。

了解如何為 RHEL 10.x 設定 NVMe over Fabrics (NVMe-oF) 主機。如需更多支援和功能信息，請參閱 "RHEL ONTAP支援與功能"。

NVMe-oF 與 RHEL 10.x 有以下已知限制：

- 這 `nvme disconnect-all`此命令會斷開根檔案系統和資料檔案系統，可能會導致系統不穩定。請勿在透過 NVMe-TCP 或 NVMe-FC 命名空間從 SAN 啟動的系統上執行此操作。

步驟 1 ：選擇性啟用 SAN 開機

您可以設定主機以使用 SAN 啟動來簡化部署並提高可擴充性。使用"互通性對照表工具"驗證您的 Linux 作業系統、主機匯流排適配器 (HBA)、HBA 韌體、HBA 啟動 BIOS 和ONTAP版本是否支援 SAN 啟動。

步驟

1. "建立 NVMe 命名空間並將其對應到主機" 。
2. 在伺服器 BIOS 中為 SAN 啟動命名空間對應到的連接埠啟用 SAN 啟動。

   如需如何啟用HBA BIOS的相關資訊、請參閱廠商專屬的文件。

3. 重新啟動主機並驗證作業系統是否已啟動並正在運行。

步驟 2：安裝 RHEL 和 NVMe 軟體並驗證您的配置

若要為 NVMe-oF 設定主機，您需要安裝主機和 NVMe 軟體包，啟用多路徑，並驗證主機 NQN 設定。

步驟

1. 在伺服器上安裝 RHEL 10.x。安裝完成後，請確認您執行的是所需的 RHEL 10.x 核心：

   ```
   uname -r
   ```

   RHEL 核心版本範例：

   ```
   6.12.0-124.8.1.el10_1.x86_64
   ```

2. 安裝「NVMe-CLI（NVMe - CLI）套件：

```
rpm -qa|grep nvme-cli
```

下面的例子展示了 `nvme-cli`軟體包版本：

```
nvme-cli-2.13-2.el10.x86_64
```

3. 安裝 `libnvme` 套件：

```
rpm -qa|grep libnvme
```

下面的例子展示了 `libnvme`軟體包版本：

```
libnvme-1.13-1.el10.x86_64
```

4. 在主機上，檢查 hostnqn 字串 `/etc/nvme/hostnqn` ：

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

下面的例子展示了 `hostnqn`版本：

```
nqn.2014-08.org.nvmexpress:uuid:4c4c4544-0056-5410-8048-c7c04f425633
```

5. 在ONTAP系統中，驗證以下資訊： `hostnqn`字串匹配 `hostnqn`ONTAP儲存系統中對應子系統的字串：

```
::> vserver nvme subsystem host show -vserver vs_coexistence_QLE2872
```

顯示範例

```
Vserver Subsystem Priority  Host NQN
------- --------- --------
-----------------------------------------------
vs_coexistence_QLE2872
               subsystem_1
                         regular   nqn.2014-
08.org.nvmexpress:uuid:4c4c4544-0056-5410-8048-c7c04f425633
               subsystem_10
                         regular   nqn.2014-
08.org.nvmexpress:uuid:4c4c4544-0056-5410-8048-c7c04f425633
               subsystem_11
                         regular   nqn.2014-
08.org.nvmexpress:uuid:4c4c4544-0056-5410-8048-c7c04f425633
```

如果 `hostnqn`字串不匹配，請使用 `vserver modify`命令來更新 `hostnqn`對應ONTAP儲存系統子系統上的字串以匹配 `hostnqn`字串來自 `/etc/nvme/hostnqn`在主機上。

步驟 3：設定 NVMe/FC 和 NVMe/TCP

使用 Broadcom/Emulex 或 Marvell/QLogic 適配器配置 NVMe/FC，或使用手動發現和連接操作來設定 NVMe/TCP。

NVMe/FC - 博通/Emulex

為 Broadcom / Emulex 介面卡設定 NVMe / FC 。

步驟

1. 確認您使用的是支援的介面卡機型：

   a. 顯示模型名稱：

   ```
   cat /sys/class/scsi_host/host*/modelname
   ```

   您應該會看到下列輸出：

   ```
   SN1700E2P
   SN1700E2P
   ```

   b. 顯示模型描述：

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

   您應該會看到類似以下範例的輸出：

   ```
   HPE SN1700E 64Gb 2p FC HBA
   HPE SN1700E 64Gb 2p FC HBA
   ```

2. 驗證您使用的是建議的Broadcom `lpfc` 韌體與收件匣驅動程式：

   a. 顯示韌體版本：

   ```
   cat /sys/class/scsi_host/host*/fwrev
   ```

   該命令返回韌體版本：

   ```
   14.4.393.25, sli-4:6:d
   14.4.393.25, sli-4:6:d
   ```

   b. 顯示收件匣驅動程式版本：

   ```
   cat /sys/module/lpfc/version
   ```

以下範例顯示了驅動程式版本：

```
0:14.4.0.9
```

如需支援的介面卡驅動程式和韌體版本的最新清單，請參閱"互通性對照表工具"。

3. 請確認 `lpfc_enable_fc4_type` 設為 `3`：

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

4. 確認您可以檢視啟動器連接埠：

```
cat /sys/class/fc_host/host*/port_name
```

您應該會看到類似如下的輸出：

```
0x10005cba2cfca7de
0x10005cba2cfca7df
```

5. 驗證啟動器連接埠是否在線上：

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

您應該會看到下列輸出：

```
Online
Online
```

6. 確認已啟用 NVMe / FC 啟動器連接埠、且目標連接埠可見：

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

顯示範例

```
NVME Initiator Enabled
XRI Dist lpfc0 Total 6144 IO 5894 ELS 250
NVME LPORT lpfc0 WWPN x10005cba2cfca7de WWNN x20005cba2cfca7de
DID x080f00 ONLINE
NVME RPORT       WWPN x2023d039eac03c33 WWNN x2021d039eac03c33
DID x082209 TARGET DISCSRVC ONLINE
NVME RPORT       WWPN x200ed039eac03c33 WWNN x200cd039eac03c33
DID x082203 TARGET DISCSRVC ONLINE
NVME RPORT       WWPN x2022d039eac03c33 WWNN x2021d039eac03c33
DID x082609 TARGET DISCSRVC ONLINE
NVME RPORT       WWPN x200dd039eac03c33 WWNN x200cd039eac03c33
DID x082604 TARGET DISCSRVC ONLINE

NVME Statistics
LS: Xmt 0000000501 Cmpl 0000000501 Abort 00000000
LS XMIT: Err 00000000  CMPL: xb 00000000 Err 00000000
Total FCP Cmpl 00000000000583b7 Issue 000000000005840d OutIO
0000000000000056
abort 0000010f noxri 00000000 nondlp 00000000 qdepth 00000000
wqerr 00000000   err 00000000
FCP CMPL: xb 0000010f Err 0000010f

NVME Initiator Enabled
XRI Dist lpfc1 Total 6144 IO 5894 ELS 250
NVME LPORT lpfc1 WWPN x10005cba2cfca7df WWNN x20005cba2cfca7df
DID x080b00 ONLINE
NVME RPORT       WWPN x2024d039eac03c33 WWNN x2021d039eac03c33
DID x082309 TARGET DISCSRVC ONLINE
NVME RPORT       WWPN x200fd039eac03c33 WWNN x200cd039eac03c33
DID x082304 TARGET DISCSRVC ONLINE
NVME RPORT       WWPN x2025d039eac03c33 WWNN x2021d039eac03c33
DID x082708 TARGET DISCSRVC ONLINE
NVME RPORT       WWPN x2010d039eac03c33 WWNN x200cd039eac03c33
DID x082703 TARGET DISCSRVC ONLINE

NVME Statistics
LS: Xmt 00000006eb Cmpl 00000006eb Abort 00000000
LS XMIT: Err 00000000  CMPL: xb 00000000 Err 00000000
Total FCP Cmpl 000000000004d600 Issue 000000000004d65f OutIO
000000000000005f
abort 000001c1 noxri 00000000 nondlp 00000000 qdepth 00000000
wqerr 00000000 err 00000000
FCP CMPL: xb 000001c1 Err 000001c2
```

NVMe/FC - Marvell/QLogic

為 Marvell/QLogic 介面卡設定 NVMe / FC 。

步驟

1. 驗證您使用的適配器驅動程式和韌體版本是否受支援：

```
cat /sys/class/fc_host/host*/symbolic_name
```

以下範例顯示了驅動程式和韌體版本：

```
QLE2872 FW:v9.15.06 DVR:v10.02.09.400-k
QLE2872 FW:v9.15.06 DVR:v10.02.09.400-k
```

2. 請確認 `ql2xnvmeenable` 已設定。這可讓 Marvell 介面卡作為 NVMe / FC 啟動器運作：

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

預期輸出為 1 。

NVMe / TCP

NVMe/TCP 協定不支援自動連線操作。相反，您可以透過執行 NVMe/TCP 來發現 NVMe/TCP 子系統和命名空間 `connect`或者 `connect-all`手動操作。

步驟

1. 檢查啟動器連接埠是否可以跨支援的 NVMe/TCP LIF 取得發現日誌頁面資料：

```
nvme discover -t tcp -w host-traddr -a traddr
```

顯示範例

```
nvme discover -t tcp -w 192.168.20.21 -a 192.168.20.28
Discovery Log Number of Records 8, Generation counter 10
=====Discovery Log Entry 0======
trtype:  tcp
adrfam:  ipv4
subtype: current discovery subsystem
treq:    not specified
portid:  8
trsvcid: 8009
subnqn:  nqn.1992-
08.com.netapp:sn.17e32b6e8c7f11f09545d039eac03c33:discovery
traddr:  192.168.21.29
eflags:  explicit discovery connections, duplicate discovery
information
sectype: none
=====Discovery Log Entry 1======
trtype:  tcp
adrfam:  ipv4
subtype: current discovery subsystem
treq:    not specified
portid:  6
trsvcid: 8009
subnqn:  nqn.1992-
08.com.netapp:sn.17e32b6e8c7f11f09545d039eac03c33:discovery
traddr:  192.168.20.29
eflags:  explicit discovery connections, duplicate discovery
information
sectype: none
=====Discovery Log Entry 2======
trtype:  tcp
adrfam:  ipv4
subtype: current discovery subsystem
treq:    not specified
portid:  7
trsvcid: 8009
subnqn:  nqn.1992-
08.com.netapp:sn.17e32b6e8c7f11f09545d039eac03c33:discovery
traddr:  192.168.21.28
eflags:  explicit discovery connections, duplicate discovery
information
sectype: none
=====Discovery Log Entry 3======
trtype:  tcp
```

```
adrfam:  ipv4
subtype: current discovery subsystem
treq:    not specified
portid:  5
trsvcid: 8009
subnqn:  nqn.1992-
08.com.netapp:sn.17e32b6e8c7f11f09545d039eac03c33:discovery
traddr:  192.168.20.28
eflags:  explicit discovery connections, duplicate discovery
information
sectype: none
=====Discovery Log Entry 4======
trtype:  tcp
adrfam:  ipv4
subtype: nvme subsystem
treq:    not specified
portid:  8
trsvcid: 4420
subnqn:  nqn.1992-
08.com.netapp:sn.17e32b6e8c7f11f09545d039eac03c33:subsystem.Bidi
rectional_DHCP_1_0
traddr:  192.168.21.29
eflags:  none
sectype: none
=====Discovery Log Entry 5======
trtype:  tcp
adrfam:  ipv4
subtype: nvme subsystem
treq:    not specified
portid:  6
trsvcid: 4420
subnqn:  nqn.1992-
08.com.netapp:sn.17e32b6e8c7f11f09545d039eac03c33:subsystem.Bidi
rectional_DHCP_1_0
traddr:  192.168.20.29
eflags:  none
sectype: none
=====Discovery Log Entry 6======
trtype:  tcp
adrfam:  ipv4
subtype: nvme subsystem
treq:    not specified
portid:  7
trsvcid: 4420
subnqn:  nqn.1992-
08.com.netapp:sn.17e32b6e8c7f11f09545d039eac03c33:subsystem.Bidi
```

```
rectional_DHCP_1_0
traddr:  192.168.21.28
eflags:  none
sectype: none
=====Discovery Log Entry 7======
trtype:  tcp
adrfam:  ipv4
subtype: nvme subsystem
treq:    not specified
portid:  5
trsvcid: 4420
subnqn:  nqn.1992-
08.com.netapp:sn.17e32b6e8c7f11f09545d039eac03c33:subsystem.Bidi
rectional_DHCP_1_0
traddr:  192.168.20.28
eflags:  none
sectype: non
```

2. 驗證其他 NVMe/TCP 啟動器-目標 LIF 組合是否可以成功檢索發現日誌頁面資料：

```
nvme discover -t tcp -w host-traddr -a traddr
```

顯示範例

```
nvme discover -t tcp -w 192.168.20.21 -a 192.168.20.28
nvme discover -t tcp -w 192.168.21.21 -a 192.168.21.28
nvme discover -t tcp -w 192.168.20.21 -a 192.168.20.29
nvme discover -t tcp -w 192.168.21.21 -a 192.168.21.29
```

3. 執行 `nvme connect-all` 跨所有節點支援的 NVMe / TCP 啟動器目標生命體執行命令：

```
nvme connect-all -t tcp -w host-traddr -a traddr
```

顯示範例

```
nvme    connect-all -t  tcp -w  192.168.20.21   -a 192.168.20.28
nvme    connect-all -t  tcp -w  192.168.21.21   -a 192.168.21.28
nvme    connect-all -t  tcp -w  192.168.20.21   -a 192.168.20.29
nvme    connect-all -t  tcp -w  192.168.21.21   -a 192.168.21.29
```

從 RHEL 9.4 開始，NVMe/TCP 的設置 `ctrl_loss_tmo timeout`自動設定為“關閉”。因此：

- 重試次數沒有限制（無限重試）。
- 您不需要手動配置特定的 `ctrl_loss_tmo timeout`使用時長 `nvme connect`或者 `nvme connect-all`命令（選項 -l ）。
- 如果發生路徑故障，NVMe/TCP 控制器不會逾時，並且會無限期地保持連線。

步驟 4：（可選）修改 udev 規則中的 iopolicy

RHEL 10.0 將 NVMe-oF 的預設 iopolicy 設定為 `round-robin`。如果您使用的是 RHEL 10.0 並且想要更改 iopolicy，請執行以下操作： `queue-depth`修改 udev 規則檔如下：

步驟

1. 使用 root 權限在文字編輯器中開啟 udev 規則檔：

   ```
   /usr/lib/udev/rules.d/71-nvmf-netapp.rules
   ```

   您應該會看到下列輸出：

   ```
   vi /usr/lib/udev/rules.d/71-nvmf-netapp.rules
   ```

2. 找到為NetApp ONTAP控制器設定 iopolicy 的行，如下例所示：

   ```
   ACTION=="add", SUBSYSTEM=="nvme-subsystem", ATTR{subsystype}=="nvm",
   ATTR{model}=="NetApp ONTAP Controller", ATTR{iopolicy}="round-robin"
   ```

3. 修改規則，以便 `round-robin`變成 `queue-depth`：

   ```
   ACTION=="add", SUBSYSTEM=="nvme-subsystem", ATTR{subsystype}=="nvm",
   ATTR{model}=="NetApp ONTAP Controller", ATTR{iopolicy}="queue-depth"
   ```

4. 重新載入udev規則並套用變更：

   ```
   udevadm control --reload
   udevadm trigger --subsystem-match=nvme-subsystem
   ```

5. 請檢查子系統的目前 I/O 策略。例如，替換<子系統> `nvme-subsys0` 。

```
cat /sys/class/nvme-subsystem/<subsystem>/iopolicy
```

您應該會看到下列輸出：

```
queue-depth.
```

> 新的 iopolicy 會自動套用於相符的NetApp ONTAP控制器設備。您無需重啟。

步驟 5：可選，啟用 NVMe/FC 的 1MB I/O。

ONTAP在識別控制器資料中報告最大資料傳輸大小 (MDTS) 為 8。這意味著最大 I/O 請求大小可達 1MB。若要向 Broadcom NVMe/FC 主機發出 1MB 大小的 I/O 要求，您應該會增加 `lpfc`的價值 `lpfc_sg_seg_cnt`參數從預設值 64 更改為 256。

> 這些步驟不適用於 Qlogic NVMe / FC 主機。

步驟

1. 將 `lpfc_sg_seg_cnt`參數設定為 256 ：

```
cat /etc/modprobe.d/lpfc.conf
```

您應該會看到類似以下範例的輸出：

```
options lpfc lpfc_sg_seg_cnt=256
```

2. 執行 `dracut -f`命令，然後重新啟動主機。
3. 確認的值 `lpfc_sg_seg_cnt`為 256 ：

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

步驟 6：驗證 NVMe 啟動服務

這 `nvmefc-boot-connections.service`和 `nvmf-autoconnect.service`NVMe/FC 中包含的啟動服務 `nvme-cli`系統啟動時，軟體包會自動啟用。

啟動完成後，驗證 `nvmefc-boot-connections.service`和 `nvmf-autoconnect.service`啟動服務已啟用。

步驟

1. 確認 `nvmf-autoconnect.service`已啟用：

```
systemctl status nvmf-autoconnect.service
```

顯示範例輸出

```
nvmf-autoconnect.service - Connect NVMe-oF subsystems automatically
during boot
     Loaded: loaded (/usr/lib/systemd/system/nvmf-
autoconnect.service; enabled; preset: disabled)
     Active: inactive (dead) since Sun 2025-10-12 19:41:15 IST; 1
day 1h ago
 Invocation: 7b5b99929c6b41199d493fa25b629f6c
   Main PID: 10043 (code=exited, status=0/SUCCESS)
   Mem peak: 2.9M
        CPU: 50ms

Oct 12 19:41:15 localhost.localdomain systemd[1]: Starting nvmf-
autoconnect.service - Connect NVMe-oF subsystems automatically
during boot...
Oct 12 19:41:15 localhost.localdomain systemd[1]: nvmf-
autoconnect.service: Deactivated successfully.
Oct 12 19:41:15 localhost.localdomain systemd[1]: Finished nvmf-
autoconnect.service - Connect NVMe-oF subsystems automatically
during boot.
```

2. 確認 `nvmefc-boot-connections.service`已啟用：

```
systemctl status nvmefc-boot-connections.service
```

顯示範例輸出

```
nvmefc-boot-connections.service - Auto-connect to subsystems on FC-
NVME devices found during boot
     Loaded: loaded (/usr/lib/systemd/system/nvmefc-boot-
connections.service; enabled; preset: enabled)
     Active: inactive (dead) since Sun 2025-10-12 19:40:33 IST; 1
day 1h ago
 Invocation: 0ec258a9f8c342ffb82408086d409bc6
   Main PID: 4151 (code=exited, status=0/SUCCESS)
   Mem peak: 2.9M
        CPU: 17ms

Oct 12 19:40:33 localhost systemd[1]: Starting nvmefc-boot-
connections.service - Auto-connect to subsystems on FC-NVME devices
found during boot...
Oct 12 19:40:33 localhost systemd[1]: nvmefc-boot-
connections.service: Deactivated successfully.
Oct 12 19:40:33 localhost systemd[1]: Finished nvmefc-boot-
connections.service - Auto-connect to subsystems on FC-NVME devices
found during boot.
```

步驟 7：驗證多路徑配置

驗證核心內建 NVMe 多重路徑狀態， ANA 狀態和 ONTAP 命名空間是否適用於 NVMe 組態。

步驟

1. 驗證主機上是否正確顯示了對應ONTAP命名空間的 NVMe-oF 設定（例如，將型號設定為NetApp ONTAP Controller，並將負載平衡 iopolicy 設定為 queue-depth）：

   a. 顯示子系統：

   ```
   cat /sys/class/nvme-subsystem/nvme-subsys*/model
   ```

   您應該會看到下列輸出：

   ```
   NetApp ONTAP Controller
   NetApp ONTAP Controller
   ```

   b. 顯示策略：

```
cat /sys/class/nvme-subsystem/nvme-subsys*/iopolicy
```

您應該會看到下列輸出：

```
queue-depth
queue-depth
```

2. 確認已在主機上建立並正確探索命名空間：

```
nvme list
```

顯示範例

```
Node                  Generic               SN
Model
--------------------- --------------------- --------------------
----------------------------------------
/dev/nvme11n1         /dev/ng11n1           81OcqJXhgWtsAAAAAAAI
NetApp ONTAP Controller

Namespace  Usage                      Format           FW Rev
---------- -------------------------- ---------------- --------
0x1        951.90  MB /  21.47  GB    4 KiB +  0 B     9.18.1
```

3. 確認每個路徑的控制器狀態均為有效、且具有正確的ANA狀態：

NVMe / FC

```
nvme list-subsys /dev/nvme9n2
```

顯示範例

```
nvme-subsys9 - NQN=nqn.1992-
08.com.netapp:sn.7c34ab26675e11f0a6c0d039eac03c33:subsystem.subs
ystem_46
               hostnqn=nqn.2014-08.org.nvmexpress:uuid:4c4c4544-
0056-5410-8048-c7c04f425633
\
 +- nvme105 fc traddr=nn-0x2018d039eac03c33:pn-
0x201bd039eac03c33,host_traddr=nn-0x2000f4c7aa0cd7c3:pn-
0x2100f4c7aa0cd7c3 live optimized
 +- nvme107 fc traddr=nn-0x2018d039eac03c33:pn-
0x2019d039eac03c33,host_traddr=nn-0x2000f4c7aa0cd7c2:pn-
0x2100f4c7aa0cd7c2 live optimized
 +- nvme42 fc traddr=nn-0x2018d039eac03c33:pn-
0x201cd039eac03c33,host_traddr=nn-0x2000f4c7aa0cd7c3:pn-
0x2100f4c7aa0cd7c3 live optimized
 +- nvme44 fc traddr=nn-0x2018d039eac03c33:pn-
0x201ad039eac03c33,host_traddr=nn-0x2000f4c7aa0cd7c2:pn-
0x2100f4c7aa0cd7c2 live optimized
```

NVMe / TCP

```
nvme list-subsys /dev/nvme4n2
```

顯示範例

```
nvme-subsys4 - NQN=nqn.1992-
08.com.netapp:sn.17e32b6e8c7f11f09545d039eac03c33:subsystem.Bidi
rectional_DHCP_1_0
               hostnqn=nqn.2014-08.org.nvmexpress:uuid:4c4c4544-
0054-5110-8039-c3c04f523034
\
 +- nvme4 tcp
traddr=192.168.20.28,trsvcid=4420,host_traddr=192.168.20.21,src_
addr=192.168.20.21 live optimized
 +- nvme5 tcp
traddr=192.168.20.29,trsvcid=4420,host_traddr=192.168.20.21,src_
addr=192.168.20.21 live optimized
 +- nvme6 tcp
traddr=192.168.21.28,trsvcid=4420,host_traddr=192.168.21.21,src_
addr=192.168.21.21 live optimized
 +- nvme7 tcp
traddr=192.168.21.29,trsvcid=4420,host_traddr=192.168.21.21,src_
addr=192.168.21.21 live optimized
```

4. 驗證NetApp外掛程式是否顯示每ONTAP 個版本名稱空間裝置的正確值：

欄位

```
nvme netapp ontapdevices -o column
```

顯示範例

```
Device            Vserver                   Subsystem
Namespace Path
---------------- -------------------------
------------------------- ------------------
/dev/nvme0n1      vs_nvme_sanboot_tcp       rhel_sanboot_tcp170
tcp_97

NSID UUID                                    Size
---- ------------------------------------- ---------
1    982c0f2a-6b8b-11f0-a6c0-d039eac03c33   322.12GB
```

JSON

```
nvme netapp ontapdevices -o json
```

顯示範例


```
{
  "ONTAPdevices":[
    {
      "Device":"/dev/nvme0n1",
      "Vserver":"vs_nvme_sanboot_tcp",
      "Subsystem":"rhel_sanboot_tcp170",
      "Namespace_Path":"tcp_97",
      "NSID":1,
      "UUID":"982c0f2a-6b8b-11f0-a6c0-d039eac03c33",
      "LBA_Size":4096,
      "Namespace_Size":322122547200,
      "UsedBytes":16285069312,
      "Version":"9.18.1"
    }
]
}
```

步驟 8：設定安全帶內身份驗證

支援透過 NVMe/TCP 在 RHEL 10.x 主機和ONTAP控制器之間進行安全的帶內身份驗證。

每個主機或控制器必須與一個 `DH-HMAC-CHAP`金鑰來設定安全身份驗證。 `DH-HMAC-CHAP`金鑰是 NVMe 主機或控制器的 NQN 與管理員配置的身份驗證金鑰的組合。若要驗證其對等端點、 NVMe 主機或控制器必須識別與對等端點相關的金鑰。

步驟

使用 CLI 或設定 JSON 檔案設定安全帶內身份驗證。如果您需要為不同的子系統指定不同的 dhchap 金鑰、則必須使用組態 JSON 檔案。

CLI

使用 CLI 設定安全的頻內驗證。

1. 取得主機 NQN ：

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

2. 為 RHEL 10.x 主機產生 dhchap 金鑰。

   以下輸出描述了 `gen-dhchap-key`命令參數：

```
nvme gen-dhchap-key -s optional_secret -l key_length {32|48|64} -m
HMAC_function {0|1|2|3} -n host_nqn
•   -s secret key in hexadecimal characters to be used to initialize
the host key
•   -l length of the resulting key in bytes
•   -m HMAC function to use for key transformation
0 = none, 1- SHA-256, 2 = SHA-384, 3=SHA-512
•   -n host NQN to use for key transformation
```

   在下列範例中、會產生一個隨機的 dhchap 金鑰、其中 HMAC 設為 3 （ SHA-512 ）。

```
nvme gen-dhchap-key -m 3 -n nqn.2014-
08.org.nvmexpress:uuid:4c4c4544-0054-5110-8039-c3c04f523034
DHHC-
1:03:AppJHkJygA6ZC4BxyQNtJST+4k4IOv47MAJk0xBITwFOHIC2nV/uE04RoSpy1z2
SXYqNW1bhLe9hJ+MDHigGexaG2Ig=:
```

3. 在 ONTAP 控制器上、新增主機並指定兩個 dhchap 金鑰：

```
vserver nvme subsystem host add -vserver <svm_name> -subsystem
<subsystem> -host-nqn <host_nqn> -dhchap-host-secret
<authentication_host_secret> -dhchap-controller-secret
<authentication_controller_secret> -dhchap-hash-function {sha-
256|sha-512} -dhchap-group {none|2048-bit|3072-bit|4096-bit|6144-
bit|8192-bit}
```

4. 主機支援兩種驗證方法：單向和雙向。在主機上、連線至 ONTAP 控制器、並根據所選的驗證方法指定 dhchap 金鑰：

```
nvme connect -t tcp -w <host-traddr> -a <tr-addr> -n <host_nqn> -S
<authentication_host_secret> -C <authentication_controller_secret>
```

5. 驗證 `nvme connect authentication` 命令驗證主機和控制器 dhchap 金鑰：
   a. 驗證主機 dhchap 金鑰：

   ```
   cat /sys/class/nvme-subsystem/<nvme-subsysX>/nvme*/dhchap_secret
   ```

   顯示單向組態的輸出範例

   ```
   cat /sys/class/nvme-subsystem/nvme-subsys4/nvme*/dhchap_secret
   DHHC-1:01:2G7lsg9PMO00h1Wf1g4QtP0XT11kREz0qVuLm2xvZdbaWR/g:
   DHHC-1:01:2G7lsg9PMO00h1Wf1g4QtP0XT11kREz0qVuLm2xvZdbaWR/g:
   DHHC-1:01:2G7lsg9PMO00h1Wf1g4QtP0XT11kREz0qVuLm2xvZdbaWR/g:
   DHHC-1:01:2G7lsg9PMO00h1Wf1g4QtP0XT11kREz0qVuLm2xvZdbaWR/g:
   ```

   b. 驗證控制器 dhchap 按鍵：

   ```
   cat /sys/class/nvme-subsystem/<nvme-
   subsysX>/nvme*/dhchap_ctrl_secret
   ```

   顯示雙向組態的輸出範例

   ```
   cat /sys/class/nvme-subsystem/nvme-
   subsys4/nvme*/dhchap_ctrl_secret
   DHHC-
   1:03:5CgWULVnU5HUOwP1MNg95pkiUAwayiO+IvrALZR8HpeJIHw3xyHdGlTnv
   EJ81HDjBb+fGteUgIn0fj8ASHZIgkuFIx8=:
   DHHC-
   1:03:5CgWULVnU5HUOwP1MNg95pkiUAwayiO+IvrALZR8HpeJIHw3xyHdGlTnv
   EJ81HDjBb+fGteUgIn0fj8ASHZIgkuFIx8=:
   DHHC-
   1:03:5CgWULVnU5HUOwP1MNg95pkiUAwayiO+IvrALZR8HpeJIHw3xyHdGlTnv
   EJ81HDjBb+fGteUgIn0fj8ASHZIgkuFIx8=:
   DHHC-
   1:03:5CgWULVnU5HUOwP1MNg95pkiUAwayiO+IvrALZR8HpeJIHw3xyHdGlTnv
   EJ81HDjBb+fGteUgIn0fj8ASHZIgkuFIx8=:
   ```

JSON

當ONTAP控制器上有多個 NVMe 子系統可用時，您可以使用 `/etc/nvme/config.json`文件與 `nvme connect-all`命令。

使用 `-o`選項來產生 JSON 檔案。有關更多語法選項，請參閱 NVMe connect-all 手冊頁。

1. 配置 JSON 檔案。

> 在以下範例中， `dhchap_key`對應於 `dhchap_secret`和 `dhchap_ctrl_key`對應於 `dhchap_ctrl_secret` 。

顯示範例


```
[
  {
    "hostnqn":"nqn.2014-08.org.nvmexpress:uuid:4c4c4544-0054-
5110-8039-c3c04f523034",
    "hostid":"44454c4c-5400-1051-8039-c3c04f523034",
    "dhchap_key":"DHHC-
1:01:2G7lsg9PMO00h1Wf1g4QtP0XT11kREz0qVuLm2xvZdbaWR/g:",
    "subsystems":[
      {
        "nqn":"nqn.1992-
08.com.netapp:sn.5857c8c9b22411f08d0ed039eac03c33:subsystem.Bidi
rectional_DHCP_1_0",
        "ports":[
          {
            "transport":"tcp",
            "traddr":"192.168.20.28",
            "host_traddr":"192.168.20.21",
            "trsvcid":"4420",
            "dhchap_ctrl_key":"DHHC-
1:03:5CgWULVnU5HUOwP1MNg95pkiUAwayiO+IvrALZR8HpeJIHw3xyHdGlTnvEJ
81HDjBb+fGteUgIn0fj8ASHZIgkuFIx8=:"
          },
          {
            "transport":"tcp",
            "traddr":"192.168.20.29",
            "host_traddr":"192.168.20.21",
            "trsvcid":"4420",
            "dhchap_ctrl_key":"DHHC-
1:03:5CgWULVnU5HUOwP1MNg95pkiUAwayiO+IvrALZR8HpeJIHw3xyHdGlTnvEJ
81HDjBb+fGteUgIn0fj8ASHZIgkuFIx8=:"
          },
          {
            "transport":"tcp",
            "traddr":"192.168.21.28",
            "host_traddr":"192.168.21.21",
            "trsvcid":"4420",
            "dhchap_ctrl_key":"DHHC-
1:03:5CgWULVnU5HUOwP1MNg95pkiUAwayiO+IvrALZR8HpeJIHw3xyHdGlTnvEJ
81HDjBb+fGteUgIn0fj8ASHZIgkuFIx8=:"
          },
          {
            "transport":"tcp",
            "traddr":"192.168.21.29",
```

```
            "host_traddr":"192.168.21.21",
            "trsvcid":"4420",
            "dhchap_ctrl_key":"DHHC-
1:03:5CgWULVnU5HUOwP1MNg95pkiUAwayiO+IvrALZR8HpeJIHw3xyHdGlTnvEJ
81HDjBb+fGteUgIn0fj8ASHZIgkuFIx8=:"
          }
        ]
      }
    ]
  }
]
```

2. 使用組態 JSON 檔案連線至 ONTAP 控制器：

```
nvme connect-all -J /etc/nvme/config.json
```

顯示範例

```
traddr=192.168.20.28 is already connected
traddr=192.168.20.28 is already connected
traddr=192.168.20.29 is already connected
traddr=192.168.20.29 is already connected
```

3. 驗證每個子系統的相應控制器是否已啟用 dhchap 機密。

   a. 驗證主機 dhchap 金鑰：

   ```
   cat /sys/class/nvme-subsystem/nvme-subsys4/nvme4/dhchap_secret
   ```

   以下範例顯示了 dhchap 金鑰：

   ```
   DHHC-1:01:2G7lsg9PMO00h1Wf1g4QtP0XT11kREz0qVuLm2xvZdbaWR/g:
   ```

   b. 驗證控制器 dhchap 按鍵：

   ```
   cat /sys/class/nvme-subsystem/nvme-
   subsys4/nvme4/dhchap_ctrl_secret
   ```

   您應該會看到類似以下範例的輸出：

```
DHHC-
1:03:5CgWULVnU5HUOwP1MNg95pkiUAwayiO+IvrALZR8HpeJIHw3xyHdGlTnvEJ81HD
jBb+fGteUgIn0fj8ASHZIgkuFIx8=:
```

步驟 9 ：檢閱已知問題

沒有已知問題。

配置 RHEL 9.x 以使用 NVMe-oF 和ONTAP存儲

Red Hat Enterpirse Linux (RHEL) 主機支援具有非對稱命名空間存取 (ANA) 的 NVMe over Fibre Channel (NVMe/FC) 和 NVMe over TCP (NVMe/TCP) 協定。 ANA 提供與 iSCSI 和 FCP 環境中的非對稱邏輯單元存取 (ALUA) 等效的多路徑功能。

了解如何為 RHEL 9.x 設定 NVMe over Fabrics (NVMe-oF) 主機。如需更多支援和功能信息，請參閱 "RHEL ONTAP支援與功能"。

NVMe-oF 與 RHEL 9.x 有以下已知限制：

- 這 `nvme disconnect-all`此命令會斷開根檔案系統和資料檔案系統，可能會導致系統不穩定。請勿在透過 NVMe-TCP 或 NVMe-FC 命名空間從 SAN 啟動的系統上執行此操作。

步驟 1 ：選擇性啟用 SAN 開機

您可以設定主機以使用 SAN 啟動來簡化部署並提高可擴充性。使用"互通性對照表工具"驗證您的 Linux 作業系統、主機匯流排適配器 (HBA)、HBA 韌體、HBA 啟動 BIOS 和ONTAP版本是否支援 SAN 啟動。

步驟

1. "建立 NVMe 命名空間並將其對應到主機" 。
2. 在伺服器 BIOS 中為 SAN 啟動命名空間對應到的連接埠啟用 SAN 啟動。

   如需如何啟用HBA BIOS的相關資訊、請參閱廠商專屬的文件。

3. 重新啟動主機並驗證作業系統是否已啟動並正在運行。

步驟 2：安裝 RHEL 和 NVMe 軟體並驗證您的配置

若要為 NVMe-oF 設定主機，您需要安裝主機和 NVMe 軟體包，啟用多路徑，並驗證主機 NQN 設定。

步驟

1. 在伺服器上安裝 RHEL 9.x。安裝完成後，請確認您執行的是所需的 RHEL 9.x 核心：

   ```
   uname -r
   ```

RHEL 核心版本範例：

```
5.14.0-611.5.1.el9_7.x86_64
```

2. 安裝「NVMe-CLI（NVMe - CLI）套件：

```
rpm -qa|grep nvme-cli
```

下面的例子展示了 `nvme-cli`軟體包版本：

```
nvme-cli-2.13-1.el9.x86_64
```

3. 安裝 `libnvme` 套件：

```
rpm -qa|grep libnvme
```

下面的例子展示了 `libnvme`軟體包版本：

```
libnvme-1.13-1.el9.x86_64
```

4. 在主機上，檢查 hostnqn 字串 `/etc/nvme/hostnqn` ：

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

下面的例子展示了 `hostnqn`版本：

```
nqn.2014-08.org.nvmexpress:uuid:4c4c4544-0056-5410-8048-b9c04f425633
```

5. 在ONTAP系統中，驗證以下資訊： `hostnqn`字串匹配 `hostnqn`ONTAP儲存系統中對應子系統的字串：

```
::> vserver nvme subsystem host show -vserver vs_188
```

顯示範例

```
Vserver Subsystem Priority  Host NQN
------- --------- --------
-----------------------------------------------
vs_188  Nvme1
                  regular   nqn.2014-
08.org.nvmexpress:uuid:4c4c4544-0056-5410-8048-b9c04f425633
        Nvme10
                  regular   nqn.2014-
08.org.nvmexpress:uuid:4c4c4544-0056-5410-8048-b9c04f425633
        Nvme11
                  regular   nqn.2014-
08.org.nvmexpress:uuid:4c4c4544-0056-5410-8048-b9c04f425633
        Nvme12
                  regular   nqn.2014-
08.org.nvmexpress:uuid:4c4c4544-0056-5410-8048-b9c04f425633
48 entries were displayed.
```

如果 `hostnqn`字串不匹配，請使用 `vserver modify`命令來更新 `hostnqn`對應ONTAP儲存系統子系統上的字串以匹配 `hostnqn`字串來自 `/etc/nvme/hostnqn`在主機上。

步驟 3：設定 NVMe/FC 和 NVMe/TCP

使用 Broadcom/Emulex 或 Marvell/QLogic 適配器配置 NVMe/FC，或使用手動發現和連接操作來設定 NVMe/TCP。

NVMe/FC - 博通/Emulex

為 Broadcom / Emulex 介面卡設定 NVMe / FC 。

步驟

1. 確認您使用的是支援的介面卡機型：

   a. 顯示模型名稱：

   ```
   cat /sys/class/scsi_host/host*/modelname
   ```

   您應該會看到下列輸出：

   ```
   LPe36002-M64
   LPe36002-M64
   ```

   b. 顯示模型描述：

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

   您應該會看到類似以下範例的輸出：

   ```
   Emulex LightPulse LPe36002-M64 2-Port 64Gb Fibre Channel Adapter
   Emulex LightPulse LPe36002-M64 2-Port 64Gb Fibre Channel Adapter
   ```

2. 驗證您使用的是建議的Broadcom `lpfc` 韌體與收件匣驅動程式：

   a. 顯示韌體版本：

   ```
   cat /sys/class/scsi_host/host*/fwrev
   ```

   該命令返回韌體版本：

   ```
   14.4.393.53, sli-4:6:d
   14.4.393.53, sli-4:6:d
   ```

   b. 顯示收件匣驅動程式版本：

   ```
   cat /sys/module/lpfc/version
   ```

以下範例顯示了驅動程式版本：

```
0:14.4.0.9
```

如需支援的介面卡驅動程式和韌體版本的最新清單，請參閱"互通性對照表工具"。

3. 請確認 `lpfc_enable_fc4_type` 設為 `3`：

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

4. 確認您可以檢視啟動器連接埠：

```
cat /sys/class/fc_host/host*/port_name
```

以下範例顯示連接埠標識：

```
0x100000109bf044b1
0x100000109bf044b2
```

5. 驗證啟動器連接埠是否在線上：

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

您應該會看到下列輸出：

```
Online
Online
```

6. 確認已啟用 NVMe / FC 啟動器連接埠、且目標連接埠可見：

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

顯示範例

```
NVME Initiator Enabled
XRI Dist lpfc0 Total 6144 IO 5894 ELS 250
NVME LPORT lpfc0 WWPN x100000109b954518 WWNN x200000109b954518
DID x020700 ONLINE
NVME RPORT       WWPN x2022d039eaa7dfc8 WWNN x201fd039eaa7dfc8
DID x020b03 TARGET DISCSRVC ONLINE
NVME RPORT       WWPN x2023d039eaa7dfc8 WWNN x201fd039eaa7dfc8
DID x020103 TARGET DISCSRVC ONLINE

NVME Statistics
LS: Xmt 0000000548 Cmpl 0000000548 Abort 00000000
LS XMIT: Err 00000000  CMPL: xb 00000000 Err 00000000
Total FCP Cmpl 0000000000001a68 Issue 0000000000001a68 OutIO
0000000000000000
        abort 00000000 noxri 00000000 nondlp 00000000 qdepth
00000000 wqerr 00000000 err 00000000
FCP CMPL: xb 00000000 Err 00000000

NVME Initiator Enabled
XRI Dist lpfc1 Total 6144 IO 5894 ELS 250
NVME LPORT lpfc1 WWPN x100000109b954519 WWNN x200000109b954519
DID x020500 ONLINE
NVME RPORT       WWPN x2027d039eaa7dfc8 WWNN x2025d039eaa7dfc8
DID x020b01 TARGET DISCSRVC ONLINE

NVME Statistics
LS: Xmt 00000005ab Cmpl 00000005ab Abort 00000000
LS XMIT: Err 00000000  CMPL: xb 00000000 Err 00000000
Total FCP Cmpl 0000000000086ce1 Issue 0000000000086ce2 OutIO
0000000000000001
        abort 0000009c noxri 00000000 nondlp 00000002 qdepth
00000000 wqerr 00000000 err 00000000
FCP CMPL: xb 000000b8 Err 000000b8

NVME Initiator Enabled
XRI Dist lpfc2 Total 6144 IO 5894 ELS 250
NVME LPORT lpfc2 WWPN x100000109bf044b1 WWNN x200000109bf044b1
DID x022a00 ONLINE
NVME RPORT       WWPN x2027d039eaa7dfc8 WWNN x2025d039eaa7dfc8
DID x020b01 TARGET DISCSRVC ONLINE
NVME RPORT       WWPN x2011d039eaa7dfc8 WWNN x200fd039eaa7dfc8
DID x020b02 TARGET DISCSRVC ONLINE
NVME RPORT       WWPN x2002d039eaa7dfc8 WWNN x2000d039eaa7dfc8
DID x020b05 TARGET DISCSRVC ONLINE
```

```
NVME RPORT       WWPN x2026d039eaa7dfc8 WWNN x2025d039eaa7dfc8
DID x021301 TARGET DISCSRVC ONLINE
NVME RPORT       WWPN x2010d039eaa7dfc8 WWNN x200fd039eaa7dfc8
DID x021302 TARGET DISCSRVC ONLINE
NVME RPORT       WWPN x2001d039eaa7dfc8 WWNN x2000d039eaa7dfc8
DID x021305 TARGET DISCSRVC ONLINE

NVME Statistics
LS: Xmt 000000c186 Cmpl 000000c186 Abort 00000000
LS XMIT: Err 00000000  CMPL: xb 00000000 Err 00000000
Total FCP Cmpl 00000000c348ca37 Issue 00000000c3344057 OutIO
ffffffffffeb7620
        abort 0000815b noxri 000018b5 nondlp 00000116 qdepth
00000000 wqerr 00000000 err 00000000
FCP CMPL: xb 0000915b Err 000c6091

NVME Initiator Enabled
XRI Dist lpfc3 Total 6144 IO 5894 ELS 250
NVME LPORT lpfc3 WWPN x100000109bf044b2 WWNN x200000109bf044b2
DID x021b00 ONLINE
NVME RPORT       WWPN x2028d039eaa7dfc8 WWNN x2025d039eaa7dfc8
DID x020101 TARGET DISCSRVC ONLINE
NVME RPORT       WWPN x2012d039eaa7dfc8 WWNN x200fd039eaa7dfc8
DID x020102 TARGET DISCSRVC ONLINE
NVME RPORT       WWPN x2003d039eaa7dfc8 WWNN x2000d039eaa7dfc8
DID x020105 TARGET DISCSRVC ONLINE
NVME RPORT       WWPN x2029d039eaa7dfc8 WWNN x2025d039eaa7dfc8
DID x022901 TARGET DISCSRVC ONLINE
NVME RPORT       WWPN x2013d039eaa7dfc8 WWNN x200fd039eaa7dfc8
DID x022902 TARGET DISCSRVC ONLINE
NVME RPORT       WWPN x2004d039eaa7dfc8 WWNN x2000d039eaa7dfc8
DID x022905 TARGET DISCSRVC ONLINE

NVME Statistics
LS: Xmt 000000c186 Cmpl 000000c186 Abort 00000000
LS XMIT: Err 00000000  CMPL: xb 00000000 Err 00000000
Total FCP Cmpl 00000000b5761af5 Issue 00000000b564b55e OutIO
ffffffffffee9a69
        abort 000083d7 noxri 000016ea nondlp 00000195 qdepth
00000000 wqerr 00000002 err 00000000
FCP CMPL: xb 000094a4 Err 000c22e7
```

NVMe/FC - Marvell/QLogic

為 Marvell/QLogic 介面卡設定 NVMe / FC 。

步驟

1. 驗證您使用的適配器驅動程式和韌體版本是否受支援：

```
cat /sys/class/fc_host/host*/symbolic_name
```

以下範例顯示了驅動程式和韌體版本：

```
QLE2872 FW:v9.15.06 DVR:v10.02.09.400-k
QLE2872 FW:v9.15.06 DVR:v10.02.09.400-k
```

2. 請確認 `ql2xnvmeenable` 已設定。這可讓 Marvell 介面卡作為 NVMe / FC 啟動器運作：

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

預期輸出為 1 。

NVMe / TCP

NVMe/TCP 協定不支援自動連線操作。相反，您可以透過執行 NVMe/TCP 來發現 NVMe/TCP 子系統和命名空間 `connect`或者 `connect-all`手動操作。

步驟

1. 檢查啟動器連接埠是否可以跨支援的 NVMe/TCP LIF 取得發現日誌頁面資料：

```
nvme discover -t tcp -w host-traddr -a traddr
```

顯示範例

```
nvme discover -t tcp -w 192.168.30.15 -a 192.168.30.48

Discovery Log Number of Records 8, Generation counter 18
=====Discovery Log Entry 0======
trtype:  tcp
adrfam:  ipv4
subtype: current discovery subsystem
treq:    not specified
portid:  8
trsvcid: 8009
subnqn:  nqn.1992-
08.com.netapp:sn.51a3c9846e0c11f08f5dd039eaa7dfc9:discovery
traddr:  192.168.31.49
eflags:  explicit discovery connections, duplicate discovery
information
sectype: none
=====Discovery Log Entry 1======
trtype:  tcp
adrfam:  ipv4
subtype: current discovery subsystem
treq:    not specified
portid:  7
trsvcid: 8009
subnqn:  nqn.1992-
08.com.netapp:sn.51a3c9846e0c11f08f5dd039eaa7dfc9:discovery
traddr:  192.168.31.48
eflags:  explicit discovery connections, duplicate discovery
information
sectype: none
=====Discovery Log Entry 2======
trtype:  tcp
adrfam:  ipv4
subtype: current discovery subsystem
treq:    not specified
portid:  6
trsvcid: 8009
subnqn:  nqn.1992-
08.com.netapp:sn.51a3c9846e0c11f08f5dd039eaa7dfc9:discovery
traddr:  192.168.30.49
eflags:  explicit discovery connections, duplicate discovery
information
sectype: none
=====Discovery Log Entry 3======
```

```
trtype:  tcp
adrfam:  ipv4
subtype: current discovery subsystem
treq:    not specified
portid:  5
trsvcid: 8009
subnqn:  nqn.1992-
08.com.netapp:sn.51a3c9846e0c11f08f5dd039eaa7dfc9:discovery
traddr:  192.168.30.48
eflags:  explicit discovery connections, duplicate discovery
information
sectype: none
=====Discovery Log Entry 4======
trtype:  tcp
adrfam:  ipv4
subtype: nvme subsystem
treq:    not specified
portid:  8
trsvcid: 4420
subnqn:  nqn.1992-
08.com.netapp:sn.51a3c9846e0c11f08f5dd039eaa7dfc9:subsystem.Nvme
38
traddr:  192.168.31.49
eflags:  none
sectype: none
=====Discovery Log Entry 5======
trtype:  tcp
adrfam:  ipv4
subtype: nvme subsystem
treq:    not specified
portid:  7
trsvcid: 4420
subnqn:  nqn.1992-
08.com.netapp:sn.51a3c9846e0c11f08f5dd039eaa7dfc9:subsystem.Nvme
38
traddr:  192.168.31.48
eflags:  none
sectype: none
=====Discovery Log Entry 6======
trtype:  tcp
adrfam:  ipv4
subtype: nvme subsystem
treq:    not specified
portid:  6
trsvcid: 4420
subnqn:  nqn.1992-
```

```
08.com.netapp:sn.51a3c9846e0c11f08f5dd039eaa7dfc9:subsystem.Nvme
38
traddr:  192.168.30.49
eflags:  none
sectype: none
=====Discovery Log Entry 7======
trtype:  tcp
adrfam:  ipv4
subtype: nvme subsystem
treq:    not specified
portid:  5
trsvcid: 4420
subnqn:  nqn.1992-
08.com.netapp:sn.51a3c9846e0c11f08f5dd039eaa7dfc9:subsystem.Nvme
38
traddr:  192.168.30.48
eflags:  none
sectype: none
```

2. 驗證其他 NVMe/TCP 啟動器-目標 LIF 組合是否可以成功檢索發現日誌頁面資料：

```
nvme discover -t tcp -w host-traddr -a traddr
```

顯示範例

```
nvme discover -t tcp -w 192.168.30.15 -a 192.168.30.48
nvme discover -t tcp -w 192.168.30.15 -a 192.168.30.49
nvme discover -t tcp -w 192.168.31.15 -a 192.168.31.48
nvme discover -t tcp -w 192.168.31.15 -a 192.168.31.49
```

3. 執行 `nvme connect-all` 跨所有節點支援的 NVMe / TCP 啟動器目標生命體執行命令：

```
nvme connect-all -t tcp -w host-traddr -a traddr
```

顯示範例

```
nvme  connect-all -t  tcp -w  192.168.30.15 -a  192.168.30.48
nvme    connect-all -t  tcp -w  192.168.30.15   -a
192.168.30.49
nvme    connect-all -t  tcp -w  192.168.31.15   -a
192.168.31.48
nvme    connect-all -t  tcp -w  192.168.31.15   -a
192.168.31.49
```

從 RHEL 9.4 開始，NVMe/TCP 的設置 `ctrl_loss_tmo timeout`自動設定為“關閉”。因此：

- 重試次數沒有限制（無限重試）。
- 您不需要手動配置特定的 `ctrl_loss_tmo timeout`使用時長 `nvme connect`或者 `nvme connect-all`命令（選項 -l ）。
- 如果發生路徑故障，NVMe/TCP 控制器不會逾時，並且會無限期地保持連線。

步驟 4：（可選）修改 udev 規則中的 iopolicy

RHEL 9.6 將 NVMe-oF 的預設 iopolicy 設定為 `round-robin`。如果您使用的是 RHEL 9.6 並且想要更改 iopolicy，請執行以下操作：`queue-depth`修改 udev 規則檔如下：

步驟

1. 使用 root 權限在文字編輯器中開啟 udev 規則檔：

   ```
   /usr/lib/udev/rules.d/71-nvmf-netapp.rules
   ```

   您應該會看到下列輸出：

   ```
   vi /usr/lib/udev/rules.d/71-nvmf-netapp.rules
   ```

2. 找到為NetApp ONTAP控制器設定 iopolicy 的行，如下例所示：

   ```
   ACTION=="add", SUBSYSTEM=="nvme-subsystem", ATTR{subsystype}=="nvm",
   ATTR{model}=="NetApp ONTAP Controller", ATTR{iopolicy}="round-robin"
   ```

3. 修改規則，以便 `round-robin`變成 `queue-depth`：

```
ACTION=="add", SUBSYSTEM=="nvme-subsystem", ATTR{subsystype}=="nvm",
ATTR{model}=="NetApp ONTAP Controller", ATTR{iopolicy}="queue-depth"
```

4. 重新載入udev規則並套用變更：

```
udevadm control --reload
udevadm trigger --subsystem-match=nvme-subsystem
```

5. 請檢查子系統的目前 I/O 策略。例如，替換<子系統> `nvme-subsys0` 。

```
cat /sys/class/nvme-subsystem/<subsystem>/iopolicy
```

您應該會看到下列輸出：

```
queue-depth.
```

> 新的 iopolicy 會自動套用於相符的NetApp ONTAP控制器設備。您無需重啟。

步驟 5：可選，啟用 NVMe/FC 的 1MB I/O。

ONTAP在識別控制器資料中報告最大資料傳輸大小 (MDTS) 為 8。這意味著最大 I/O 請求大小可達 1MB。若要向 Broadcom NVMe/FC 主機發出 1MB 大小的 I/O 要求，您應該會增加 `lpfc`的價值 `lpfc_sg_seg_cnt`參數從預設值 64 更改為 256。

> 這些步驟不適用於 Qlogic NVMe / FC 主機。

步驟

1. 將 `lpfc_sg_seg_cnt`參數設定為 256 ：

```
cat /etc/modprobe.d/lpfc.conf
```

您應該會看到類似以下範例的輸出：

```
options lpfc lpfc_sg_seg_cnt=256
```

2. 執行 `dracut -f`命令，然後重新啟動主機。
3. 確認的值 `lpfc_sg_seg_cnt`為 256 ：

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

步驟 6：驗證 NVMe 啟動服務

這 `nvmefc-boot-connections.service`和 `nvmf-autoconnect.service`NVMe/FC 中包含的啟動服務 `nvme-cli`系統啟動時，軟體包會自動啟用。

啟動完成後，驗證 `nvmefc-boot-connections.service`和 `nvmf-autoconnect.service`啟動服務已啟用。

步驟

1. 確認 `nvmf-autoconnect.service`已啟用：

```
systemctl status nvmf-autoconnect.service
```

顯示範例輸出

```
nvmf-autoconnect.service - Connect NVMe-oF subsystems automatically
during boot
        Loaded: loaded (/usr/lib/systemd/system/nvmf-
autoconnect.service;  enabled; preset: disabled)

Active: inactive (dead) since Wed 2025-10-29 00:42:03 EDT; 6h ago
Main PID: 8487 (code=exited, status=0/SUCCESS)  CPU: 66ms

Oct 29 00:42:03 R650-14-188 systemd[1]: Starting Connect NVMe-oF
subsystems automatically during boot...
Oct 29 00:42:03 R650-14-188 systemd[1]: nvmf-autoconnect.service:
Deactivated successfully.
Oct 29 00:42:03 R650-14-188 systemd[1]: Finished Connect NVMe-oF
subsystems automatically during boot.
```

2. 確認 `nvmefc-boot-connections.service`已啟用：

```
systemctl status nvmefc-boot-connections.service
```

顯示範例輸出

```
nvmefc-boot-connections.service - Auto-connect to subsystems on FC-
NVME devices found during boot
         Loaded: loaded (/usr/lib/systemd/system/nvmefc-boot-
connections.service; enabled; preset:enabled)
     Active: inactive (dead) since Wed 2025-10-29 00:41:51 EDT; 6h
ago
Main PID: 4652 (code=exited, status=0/SUCCESS)
        CPU: 13ms

Oct 29 00:41:51 R650-14-188 systemd[1]: Starting Auto-connect to
subsystems on FC-NVME devices found during boot...  Oct 29 00:41:51
R650-14-188 systemd[1]: nvmefc-boot-connections.service: Deactivated
successfully.  Oct 29 00:41:51 R650-14-188 systemd[1]: Finished
Auto-connect to subsystems on FC-NVME devices found during boot
```

步驟 7：驗證多路徑配置

驗證核心內建 NVMe 多重路徑狀態， ANA 狀態和 ONTAP 命名空間是否適用於 NVMe 組態。

步驟

1. 確認已啟用核心內建 NVMe 多重路徑：

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

您應該會看到下列輸出：

```
Y
```

2. 驗證主機上是否正確顯示了對應ONTAP命名空間的 NVMe-oF 設定（例如，型號設定為NetApp ONTAP Controller，負載平衡 iopolicy 設定為 round-robin）：

   a. 顯示子系統：

```
cat /sys/class/nvme-subsystem/nvme-subsys*/model
```

   您應該會看到下列輸出：

```
NetApp ONTAP Controller
NetApp ONTAP Controller
```

b. 顯示策略：

```
cat /sys/class/nvme-subsystem/nvme-subsys*/iopolicy
```

您應該會看到下列輸出：

```
queue-depth
queue-depth
```

3. 確認已在主機上建立並正確探索命名空間：

```
nvme list
```

顯示範例

```
Node                                  Generic              SN
Model
----------------------------------------------------------------------
------------------
/dev/nvme100n1  /dev/ng100n1  81LJCJYaKOHhAAAAAAAf   NetApp ONTAP
Controller
Namespace Usage    Format              FW              Rev
-------------------------------------------------------------
0x1                 1.19  GB /   5.37  GB   4 KiB + 0 B   9.18.1
```

4. 確認每個路徑的控制器狀態均為有效、且具有正確的ANA狀態：

NVMe / FC

```
nvme list-subsys /dev/nvme100n1
```

顯示範例

```
nvme-subsys4 - NQN=nqn.1992-
08.com.netapp:sn.3623e199617311f09257d039eaa7dfc9:subsystem.Nvme
31
               hostnqn=nqn.2014-08.org.nvmexpress:uuid:
4c4c4544-0056-5410-8048-b9c04f42563
               \
+- nvme199 fc   traddr=nn-0x200fd039eaa7dfc8:pn-
0x2010d039eaa7dfc8,host_traddr=nn-0x200000109bf044b1:pn-
0x100000109bf044b1 live optimized
+- nvme246 fc  traddr=nn-0x200fd039eaa7dfc8:pn-
0x2011d039eaa7dfc8,host_traddr=nn-0x200000109bf044b1:pn-
0x100000109bf044b1  live non-optimized
+- nvme249 fc  traddr=nn-0x200fd039eaa7dfc8:pn-
0x2013d039eaa7dfc8,host_traddr=nn-0x200000109bf044b2:pn-
0x100000109bf044b2 live optimized
+- nvme251 fc   traddr=nn-0x200fd039eaa7dfc8:pn-
0x2012d039eaa7dfc8,host_traddr=nn-0x200000109bf044b2:pn-
0x100000109bf044b2 live non-optimized
```

NVMe / TCP

```
nvme list-subsys /dev/nvme0n1
```

顯示範例

```
nvme-subsys0 - NQN=nqn.1992-
08.com.netapp:sn.51a3c9846e0c11f08f5dd039eaa7dfc9:subsystem.Nvme
1
hostnqn=nqn.2014-08.org.nvmexpress:uuid:4c4c4544-0035-5910-804b-
b5c04f444d33
\
+- nvme0 tcp
traddr=192.168.30.48,trsvcid=4420,host_traddr=192.168.30.15,
src_addr=192.168.30.15 live optimized
+- nvme1 tcp
traddr=192.168.30.49,trsvcid=4420,host_traddr=192.168.30.15,
src_addr=192.168.30.15 live non-optimized
+- nvme2 tcp
traddr=192.168.31.48,trsvcid=4420,host_traddr=192.168.31.15,
src_addr=192.168.31.15 live optimized
+- nvme3 tcp
traddr=192.168.31.49,trsvcid=4420,host_traddr=192.168.31.15,
src_addr=192.168.31.15 live non-optimized
```

5. 驗證NetApp外掛程式是否顯示每ONTAP 個版本名稱空間裝置的正確值：

欄位

```
nvme netapp ontapdevices -o column
```

顯示範例

```
Device          Vserver         Subsystem   Namespace Path
NSID
------------    --------        ----------- -----------------
----
/dev/nvme0n1    vs_iscsi_tcp    Nvme1       /vol/Nvmevol1/ns1
1
UUID                                        Size
--------------------------------------      -----
d8efef7d-4dde-447f-b50e-b2c009298c66        26.84GB
```

JSON

```
nvme netapp ontapdevices -o json
```

顯示範例

```
{
  "ONTAPdevices":[
    {
      "Device":"/dev/nvme0n1",
      "Vserver":"vs_iscsi_tcp",
      " Subsystem":"Nvme1",
      "Namespace_Path":"/vol/Nvmevol1/ns1",
      "NSID":1,
      "UUID":"d8efef7d-4dde-447f-b50e-b2c009298c66",
      "LBA_Size":4096,
      "Namespace_Size":26843545600,
    },
]
}
```

步驟 8：設定安全帶內身份驗證

支援透過 NVMe/TCP 在 RHEL 9.x 主機和ONTAP控制器之間進行安全的帶內身份驗證。

每個主機或控制器必須與一個 `DH-HMAC-CHAP`金鑰來設定安全身份驗證。 `DH-HMAC-CHAP`金鑰是 NVMe 主機或控制器的 NQN 與管理員配置的身份驗證金鑰的組合。若要驗證其對等端點、 NVMe 主機或控制器必須識別與對等端點相關的金鑰。

步驟

使用 CLI 或設定 JSON 檔案設定安全帶內身份驗證。如果您需要為不同的子系統指定不同的 dhchap 金鑰、則必須使用組態 JSON 檔案。

CLI

使用 CLI 設定安全的頻內驗證。

1. 取得主機 NQN ：

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

2. 為 RHEL 9.x 主機產生 dhchap 金鑰。

   以下輸出描述了 `gen-dhchap-key`命令參數：

```
nvme gen-dhchap-key -s optional_secret -l key_length {32|48|64} -m
HMAC_function {0|1|2|3} -n host_nqn
•   -s secret key in hexadecimal characters to be used to initialize
the host key
•   -l length of the resulting key in bytes
•   -m HMAC function to use for key transformation
0 = none, 1- SHA-256, 2 = SHA-384, 3=SHA-512
•   -n host NQN to use for key transformation
```

   在下列範例中、會產生一個隨機的 dhchap 金鑰、其中 HMAC 設為 3 （ SHA-512 ）。

```
nvme gen-dhchap-key -m 3 -n nqn.2014-
08.org.nvmexpress:uuid:e6dade64-216d-11ec-b7bb-7ed30a5482c3
DHHC-
1:03:wSpuuKbBHTzC0W9JZxMBsYd9JFV8Si9aDh22k2BR/4m852vH7KGlrJeMpzhmyjD
WOo0PJJM6yZsTeEpGkDHMHQ255+g=:
```

3. 在 ONTAP 控制器上、新增主機並指定兩個 dhchap 金鑰：

```
vserver nvme subsystem host add -vserver <svm_name> -subsystem
<subsystem> -host-nqn <host_nqn> -dhchap-host-secret
<authentication_host_secret> -dhchap-controller-secret
<authentication_controller_secret> -dhchap-hash-function {sha-
256|sha-512} -dhchap-group {none|2048-bit|3072-bit|4096-bit|6144-
bit|8192-bit}
```

4. 主機支援兩種驗證方法：單向和雙向。在主機上、連線至 ONTAP 控制器、並根據所選的驗證方法指定 dhchap 金鑰：

```
nvme connect -t tcp -w <host-traddr> -a <tr-addr> -n <host_nqn> -S
<authentication_host_secret> -C <authentication_controller_secret>
```

5. 驗證 `nvme connect authentication` 命令驗證主機和控制器 dhchap 金鑰：
   a. 驗證主機 dhchap 金鑰：

   ```
   cat /sys/class/nvme-subsystem/<nvme-subsysX>/nvme*/dhchap_secret
   ```

   顯示單向組態的輸出範例

   ```
   cat /sys/class/nvme-subsystem/nvme-subsys1/nvme*/dhchap_secret
   DHHC-1:01:hhdIYK7rGxHiNYS4d421GxHeDRUAuY0vmdqCp/NOaYND2PSc:
   DHHC-1:01:hhdIYK7rGxHiNYS4d421GxHeDRUAuY0vmdqCp/NOaYND2PSc:
   DHHC-1:01:hhdIYK7rGxHiNYS4d421GxHeDRUAuY0vmdqCp/NOaYND2PSc:
   DHHC-1:01:hhdIYK7rGxHiNYS4d421GxHeDRUAuY0vmdqCp/NOaYND2PSc:
   ```

   b. 驗證控制器 dhchap 按鍵：

   ```
   cat /sys/class/nvme-subsystem/<nvme-
   subsysX>/nvme*/dhchap_ctrl_secret
   ```

   顯示雙向組態的輸出範例

   ```
   cat /sys/class/nvme-subsystem/nvme-
   subsys*/nvme*/dhchap_ctrl_secret

   DHHC-
   1:03:ZCRrP9MQOeXhFitT7Fvvf/3P6K/qY1HfSmSfM8nLjESJdOjbjK/J6m00y
   gJgjm0VrRlrgrnHzjtWJmsnoVBO3rPDGEk=:
   DHHC-
   1:03:ZCRrP9MQOeXhFitT7Fvvf/3P6K/qY1HfSmSfM8nLjESJdOjbjK/J6m00y
   gJgjm0VrRlrgrnHzjtWJmsnoVBO3rPDGEk=:
   DHHC-
   1:03:ZCRrP9MQOeXhFitT7Fvvf/3P6K/qY1HfSmSfM8nLjESJdOjbjK/J6m00y
   gJgjm0VrRlrgrnHzjtWJmsnoVBO3rPDGEk=:
   DHHC-
   1:03:ZCRrP9MQOeXhFitT7Fvvf/3P6K/qY1HfSmSfM8nLjESJdOjbjK/J6m00y
   gJgjm0VrRlrgrnHzjtWJmsnoVBO3rPDGEk=:
   ```

JSON

當ONTAP控制器上有多個 NVMe 子系統可用時，您可以使用 `/etc/nvme/config.json`文件與 `nvme connect-all`命令。

使用 `-o`選項來產生 JSON 檔案。有關更多語法選項，請參閱 NVMe connect-all 手冊頁。

1. 配置 JSON 檔案。

> 在以下範例中， `dhchap_key`對應於 `dhchap_secret`和 `dhchap_ctrl_key`對應於 `dhchap_ctrl_secret` 。

顯示範例

```
cat /etc/nvme/config.json
[
{
  "hostnqn":"nqn.2014-08.org.nvmexpress:uuid:4c4c4544-0035-5910-
804b-b5c04f444d33",
  "hostid":"4c4c4544-0035-5910-804b-b5c04f444d33",
  "dhchap_key":"DHHC-
1:01:GhgaLS+0h0W/IxKhSa0iaMHg17SOHRTzBduPzoJ6LKEJs3/f:",
  "subsystems":[
        {
          "nqn":"nqn.1992-
08.com.netapp:sn.2c0c80d9873a11f0bc60d039eab6cb6d:subsystem.istp
MNTC_subsys",
          "ports":[
              {
                  "transport":"tcp",
                    "traddr":"192.168.30.44",
                  "host_traddr":"192.168.30.15",
                  "trsvcid":"4420",
                  "dhchap_ctrl_key":"DHHC-
1:03:GaraCO84o/uM0jF4rKJlgTy22bVoV0dRn1M+9QDfQRNVwJDHfPu2LrK5Y+/
XG8iGcRtBCdm3
fYm3ZmO6NiepCORoY5Q=:"
              },
              {
                  "transport":"tcp",
                  "traddr":"192.168.30.45"
                  "host_traddr":"192.168.30.15",
                  "trsvcid":"4420",
                  "dhchap_ctrl_key":"DHHC-
1:03:GaraCO84o/uM0jF4rKJlgTy22bVoV0dRn1M+9QDfQRNVwJDHfPu2LrK5Y+/
XG8iGcRtBCdm3
fYm3ZmO6NiepCORoY5Q=:"
              },
              {
                  "transport":"tcp",
                 "traddr":"192.168.31.44",
                  "host_traddr":"192.168.31.15",
                  "trsvcid":"4420",
                  "dhchap_ctrl_key":"DHHC-
                  1:03:
GaraCO84o/uM0jF4rKJlgTy22bVoV0dRn1M+9QDfQRNVwJDHfPu2LrK5Y+/XG8iG
c
```

```
RtBCdm3fYm3ZmO6NiepCORoY5Q=:"                                    },
 {
                  "transport":"tcp",
                  "traddr":"192.168.31.45",
                  "host_traddr":"192.168.31.15",
                  "trsvcid":"4420",
                  "dhchap_ctrl_key":"DHHC-
                  1:03:
GaraCO84o/uM0jF4rKJlgTy22bVoV0dRn1M+9QDfQRNVwJDHfPu2LrK5Y+/XG8iG
cRtBCdm3fYm3ZmO6NiepCORoY5Q=:"
              }
        ]
  ]
}
]
```

2. 使用組態 JSON 檔案連線至 ONTAP 控制器：

```
nvme connect-all -J /etc/nvme/config.json
```

顯示範例

```
already connected to hostnqn=nqn.2014-
08.org.nvmexpress:uuid:4c4c4544-0035-5910-804b-
b5c04f444d33,nqn=nqn.1992-
08.com.netapp:sn.2c0c80d9873a11f0bc60d039eab6cb6d:subsystem.istp
MNTC_subsys,transport=tcp,traddr=192.168.30.44,trsvcid=4420
already connected to hostnqn=nqn.2014-
08.org.nvmexpress:uuid:4c4c4544-0035-5910-804b-
b5c04f444d33,nqn=nqn.1992-
08.com.netapp:sn.2c0c80d9873a11f0bc60d039eab6cb6d:subsystem.istp
MNTC_subsys,transport=tcp,traddr=192.168.31.44,trsvcid=4420
already connected to hostnqn=nqn.2014-
08.org.nvmexpress:uuid:4c4c4544-0035-5910-804b-
b5c04f444d33,nqn=nqn.1992-
08.com.netapp:sn.2c0c80d9873a11f0bc60d039eab6cb6d:subsystem.istp
MNTC_subsys,transport=tcp,traddr=192.168.30.45,trsvcid=4420
already connected to hostnqn=nqn.2014-
08.org.nvmexpress:uuid:4c4c4544-0035-5910-804b-
b5c04f444d33,nqn=nqn.1992-
08.com.netapp:sn.2c0c80d9873a11f0bc60d039eab6cb6d:subsystem.istp
MNTC_subsys,transport=tcp,traddr=192.168.31.45,trsvcid=4420
```

3. 確認已為每個子系統的個別控制器啟用 dhchap 機密：

   a. 驗證主機 dhchap 金鑰：

   ```
   cat /sys/class/nvme-subsystem/nvme-subsys96/nvme96/dhchap_secret
   ```

   以下範例顯示了 dhchap 金鑰：

   ```
   DHHC-1:01:hhdIYK7rGxHiNYS4d421GxHeDRUAuY0vmdqCp/NOaYND2PSc:
   ```

   b. 驗證控制器 dhchap 按鍵：

   ```
   cat /sys/class/nvme-subsystem/nvme-
   subsys96/nvme96/dhchap_ctrl_secret
   ```

   您應該會看到類似以下範例的輸出：

```
DHHC-
1:03:ZCRrP9MQOeXhFitT7Fvvf/3P6K/qY1HfSmSfM8nLjESJdOjbjK/J6m00ygJgjm0
VrRlrgrnHzjtWJmsnoVBO3rPDGEk=:
```

步驟 9 ：檢閱已知問題

這些是已知問題：

NetApp錯誤ID	標題	說明
1503468	在 RHEL 9.1 中， `nvme list-subsys`該命令傳回給定子系統的重複 NVMe 控制器列表	這 `nvme list-subsys`指令傳回給定子系統的 NVMe 控制器清單。在 RHEL 9.1 中，此指令顯示子系統中所有命名空間的控制器及其 ANA 狀態。由於 ANA 狀態是每個命名空間的屬性，因此此指令應顯示具有每個命名空間的路徑狀態的唯一控制器條目。
"1479047"	RHEL 9.0 NVMe-oF 主機建立重複的持久發現控制器 (PDC)	在 NVMe-oF 主機上，您可以使用 `nvme discover -p` 指令來建立 PDC。使用此指令時，每個發起方-目標組合只能建立一個 PDC。但是，如果您在執行 ONTAP 9.10.1 並搭配 RHEL 9.0 及 NVMe-oF 主機時，每次執行 `nvme discover -p` 都會建立重複的 PDC。這會導致主機和目標裝置上的資源被不必要地消耗掉。

配置 RHEL 8.x 以使用 NVMe-oF 和ONTAP存儲

Red Hat Enterpirse Linux (RHEL) 主機支援具有非對稱命名空間存取 (ANA) 的 NVMe over Fibre Channel (NVMe/FC) 和 NVMe over TCP (NVMe/TCP) 協定。 ANA 提供與 iSCSI 和 FCP 環境中的非對稱邏輯單元存取 (ALUA) 等效的多路徑功能。

了解如何為 RHEL 8.x 設定 NVMe over Fabrics (NVMe-oF) 主機。如需更多支援和功能信息，請參閱 "RHEL ONTAP支援與功能"。

NVMe-oF 與 RHEL 8.x 有以下已知限制：

- 目前不支援使用 NVMe-oF 協定的 SAN 啟動。
- 在 RHEL 8.x 的 NVMe-oF 主機上，核心內 NVMe 多路徑預設為停用；您必須手動啟用它。
- 由於已知問題，NVMe/TCP 可作為技術預覽。

步驟 1 ：選擇性啟用 SAN 開機

您可以設定主機以使用 SAN 啟動來簡化部署並提高可擴充性。使用"互通性對照表工具"驗證您的 Linux 作業系統、主機匯流排適配器 (HBA)、HBA 韌體、HBA 啟動 BIOS 和ONTAP版本是否支援 SAN 啟動。

步驟

1. "建立 NVMe 命名空間並將其對應到主機" 。
2. 在伺服器 BIOS 中為 SAN 啟動命名空間對應到的連接埠啟用 SAN 啟動。

   如需如何啟用HBA BIOS的相關資訊、請參閱廠商專屬的文件。

3. 重新啟動主機並驗證作業系統是否已啟動並正在運行。

步驟 2：安裝 RHEL 和 NVMe 軟體並驗證您的配置

若要為 NVMe-oF 設定主機，您需要安裝主機和 NVMe 軟體包，啟用多路徑，並驗證主機 NQN 設定。

步驟

1. 在伺服器上安裝 RHEL 8.x。安裝完成後，請確認您執行的是所需的 RHEL 8.x 核心：

   ```
   uname -r
   ```

   RHEL 核心版本範例：

   ```
   4.18.0-553.el8_10.x86_64
   ```

2. 安裝「NVMe-CLI（NVMe - CLI）套件：

```
rpm -qa|grep nvme-cli
```

以下範例顯示了 nvme-cli 軟體包版本：

```
nvme-cli-1.16-9.el8.x86_64
```

3. 安裝 `libnvme` 套件：

```
rpm -qa|grep libnvme
```

以下範例顯示了 libnvme 軟體包版本：

```
libnvme-1.4-3.el8.x86_64
```

4. 啟用核心內建NVMe多重路徑：

```
grubby --args=nvme_core.multipath=Y --update-kernel /boot/vmlinuz-
4.18.0-553.el8_10.x86_64
```

5. 在 RHEL 8.x 主機上，檢查 `hostnqn`**字串** `/etc/nvme/hostnqn`：

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

下面的例子展示了 `hostnqn`版本：

```
nqn.2014-08.org.nvmexpress:uuid:4c4c4544-0032-3410-8035-b8c04f4c5132
```

6. 在ONTAP系統中，驗證以下資訊：`hostnqn`字串匹配 `hostnqn`ONTAP儲存系統中對應子系統的字串：

```
::> vserver nvme subsystem host show -vserver vs_fcnvme_141
```

顯示範例

```
Vserver     Subsystem          Host NQN
----------- ---------------
------------------------------------------------------------
vs_25_2742   rhel_101_QLe2772    nqn.2014-
08.org.nvmexpress:uuid:546399fc-160f-11e5-89aa-98be942440ca
```

如果 `hostnqn`字串不匹配，請使用 `vserver modify`命令來更新 `hostnqn`對應ONTAP儲存系統子系統上的字串以匹配 `hostnqn`字串來自 `/etc/nvme/hostnqn`在主機上。

7. 重新啟動主機。

為了在同一台主機上執行 NVMe 和 SCSI 流量， NetApp建議對ONTAP命名空間使用核心 NVMe 多路徑，對ONTAP LUN 使用 dm-multipath。為了防止 dm-multipath 聲明ONTAP命名空間設備，請透過添加 `enable_foreign`設定為 `/etc/multipath.conf`文件：

```
cat /etc/multipath.conf
defaults {
        enable_foreign     NONE
}
```

步驟 3：設定 NVMe/FC 和 NVMe/TCP

使用 Broadcom/Emulex 或 Marvell/QLogic 適配器配置 NVMe/FC，或使用手動發現和連接操作來設定 NVMe/TCP。

NVMe/FC - 博通/Emulex

為 Broadcom / Emulex 介面卡設定 NVMe / FC 。

步驟

1. 確認您使用的是支援的介面卡機型：

   a. 顯示模型名稱：

   ```
   cat /sys/class/scsi_host/host*/modelname
   ```

   您應該會看到下列輸出：

   ```
   LPe32002-M2
   LPe32002-M2
   ```

   b. 顯示模型描述：

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

   您應該會看到類似以下範例的輸出：

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

2. 驗證您使用的是建議的Broadcom `lpfc` 韌體與收件匣驅動程式：

   a. 顯示韌體版本：

   ```
   cat /sys/class/scsi_host/host*/fwrev
   ```

   該命令返回韌體版本：

   ```
   14.2.539.21, sli-4:2:c
   14.2.539.21, sli-4:2:c
   ```

   b. 顯示收件匣驅動程式版本：

   ```
   cat /sys/module/lpfc/version
   ```

以下範例顯示了驅動程式版本：

```
0:14.0.0.21
```

如需支援的介面卡驅動程式和韌體版本的最新清單，請參閱"互通性對照表工具"。

3. 請確認 `lpfc_enable_fc4_type` 設為 `3`：

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

4. 確認您可以檢視啟動器連接埠：

```
cat /sys/class/fc_host/host*/port_name
```

您應該會看到類似如下的輸出：

```
0x10000090fae0ec88
0x10000090fae0ec89
```

5. 驗證啟動器連接埠是否在線上：

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

您應該會看到下列輸出：

```
Online
Online
```

6. 確認已啟用 NVMe / FC 啟動器連接埠、且目標連接埠可見：

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

顯示範例

```
NVME Initiator Enabled
XRI Dist lpfc0 Total 6144 IO 5894 ELS 250
NVME LPORT lpfc0 WWPN x100000109bf044b1 WWNN x200000109bf044b1
DID x022a00 ONLINE
NVME RPORT       WWPN x211ad039eaa7dfc8 WWNN x2119d039eaa7dfc8
DID x021302 TARGET DISCSRVC ONLINE
NVME RPORT       WWPN x211cd039eaa7dfc8 WWNN x2119d039eaa7dfc8
DID x020b02 TARGET DISCSRVC ONLINE

NVME Statistics
LS: Xmt 00000001ff Cmpl 00000001ff Abort 00000000
LS XMIT: Err 00000000  CMPL: xb 00000000 Err 00000000
Total FCP Cmpl 0000000001330ec7 Issue 0000000001330ec9 OutIO
0000000000000002
        abort 00000330 noxri 00000000 nondlp 0000000b qdepth
00000000 wqerr 00000000 err 00000000
FCP CMPL: xb 00000354 Err 00000361

NVME Initiator Enabled
XRI Dist lpfc1 Total 6144 IO 5894 ELS 250
NVME LPORT lpfc1 WWPN x100000109bf044b2 WWNN x200000109bf044b2
DID x021b00 ONLINE
NVME RPORT       WWPN x211bd039eaa7dfc8 WWNN x2119d039eaa7dfc8
DID x022902 TARGET DISCSRVC ONLINE
NVME RPORT       WWPN x211dd039eaa7dfc8 WWNN x2119d039eaa7dfc8
DID x020102 TARGET DISCSRVC ONLINE

NVME Statistics
LS: Xmt 00000001ff Cmpl 00000001ff Abort 00000000
LS XMIT: Err 00000000  CMPL: xb 00000000 Err 00000000
Total FCP Cmpl 00000000012ec220 Issue 00000000012ec222 OutIO
0000000000000002
        abort 0000033b noxri 00000000 nondlp 00000085 qdepth
00000000 wqerr 00000000 err 00000000
FCP CMPL: xb 00000368 Err 00000382
```

NVMe/FC - Marvell/QLogic

為 Marvell/QLogic 介面卡設定 NVMe / FC 。

步驟

1. 驗證您使用的適配器驅動程式和韌體版本是否受支援：

```
cat /sys/class/fc_host/host*/symbolic_name
```

以下範例顯示了驅動程式和韌體版本：

```
QLE2742 FW: v9.10.11 DVR: v10.02.08.200-k
QLE2742 FW: v9.10.11 DVR: v10.02.08.200-k
```

2. 請確認 `ql2xnvmeenable` 已設定。這可讓 Marvell 介面卡作為 NVMe / FC 啟動器運作：

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

預期輸出為 1 。

NVMe / TCP

NVMe/TCP 協定不支援自動連線操作。相反，您可以透過執行 NVMe/TCP 來發現 NVMe/TCP 子系統和命名空間 `connect`或者 `connect-all`手動操作。

步驟

1. 檢查啟動器連接埠是否可以跨支援的 NVMe/TCP LIF 取得發現日誌頁面資料：

```
nvme discover -t tcp -w host-traddr -a traddr
```

顯示範例

```
nvme discover -t tcp -w 192.168.2.31 -a 192.168.2.25

Discovery Log Number of Records 8, Generation counter 18
=====Discovery Log Entry 0======
trtype:  tcp
adrfam:  ipv4
subtype: unrecognized
treq:    not specified.
portid:  0
trsvcid: 8009
subnqn:  nqn.1992-
08.com.netapp:sn.a1b2b785b9de11ee8e7fd039ea9e8ae9:discovery
traddr:  192.168.1.25
sectype: none
=====Discovery Log Entry 1======
trtype:  tcp
adrfam:  ipv4
subtype: unrecognized
treq:    not specified.
portid:  1
trsvcid: 8009
subnqn:  nqn.1992-
08.com.netapp:sn.a1b2b785b9de11ee8e7fd039ea9e8ae9:discovery
traddr:  192.168.2.26
sectype: none
```

2. 確認其他的 NVMe / TCP 啟動器目標 LIF 組合可以成功擷取探索記錄頁面資料：

```
nvme discover -t tcp -w host-traddr -a traddr
```

顯示範例

```
nvme  discover  -t  tcp  -w 192.168.2.31  -a 192.168.2.25
nvme  discover  -t  tcp  -w 192.168.1.31  -a 192.168.1.24
nvme  discover  -t  tcp  -w 192.168.2.31  -a 192.168.2.26
nvme  discover  -t  tcp  -w 192.168.1.31  -a 192.168.1.25
```

3. 執行 `nvme connect-all` 跨所有節點支援的 NVMe / TCP 啟動器目標生命體執行命令：

```
nvme connect-all -t tcp -w host-traddr -a traddr
```

顯示範例

```
nvme connect-all -t tcp -w 192.168.2.31 -a 192.168.2.25 -l 1800
nvme connect-all -t tcp -w 192.168.1.31 -a 192.168.1.24 -l 1800
nvme connect-all -t tcp -w 192.168.2.31 -a 192.168.2.26 -l 1800
nvme connect-all -t tcp -w 192.168.1.31 -a 192.168.1.25 -l 1800
```

步驟 4：（可選）為 NVMe/FC 啟用 1MB I/O

ONTAP在識別控制器資料中報告最大資料傳輸大小 (MDTS) 為 8。這意味著最大 I/O 請求大小可達 1MB。若要向 Broadcom NVMe/FC 主機發出 1MB 大小的 I/O 要求，您應該會增加 `lpfc`的價值 `lpfc_sg_seg_cnt`參數從預設值 64 更改為 256。

> 這些步驟不適用於 Qlogic NVMe / FC 主機。

步驟

1. 將 `lpfc_sg_seg_cnt`參數設定為 256 ：

   ```
   cat /etc/modprobe.d/lpfc.conf
   ```

   您應該會看到類似以下範例的輸出：

   ```
   options lpfc lpfc_sg_seg_cnt=256
   ```

2. 執行 `dracut -f`命令，然後重新啟動主機。
3. 確認的值 `lpfc_sg_seg_cnt`為 256 ：

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

步驟 5：驗證多路徑配置

驗證核心內建 NVMe 多重路徑狀態， ANA 狀態和 ONTAP 命名空間是否適用於 NVMe 組態。

步驟

1. 確認已啟用核心內建 NVMe 多重路徑：

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

您應該會看到下列輸出：

```
Y
```

2. 驗證個別 ONTAP 命名空間的適當 NVMe 設定（例如、模型設定為 NetApp ONTAP 控制器、負載平衡 iopolicing 設定為循環）是否正確反映在主機上：

   a. 顯示子系統：

   ```
   cat /sys/class/nvme-subsystem/nvme-subsys*/model
   ```

   您應該會看到下列輸出：

   ```
   NetApp ONTAP Controller
   NetApp ONTAP Controller
   ```

   b. 顯示策略：

   ```
   cat /sys/class/nvme-subsystem/nvme-subsys*/iopolicy
   ```

   您應該會看到下列輸出：

```
round-robin
round-robin
```

3. 確認已在主機上建立並正確探索命名空間：

```
nvme list
```

顯示範例

```
Node         SN                   Model
------------------------------------------------------------
/dev/nvme4n1 81Ix2BVuekWcAAAAAAAB   NetApp ONTAP Controller


Namespace Usage    Format             FW             Rev
-----------------------------------------------------------
1                 21.47 GB / 21.47 GB   4 KiB + 0 B   FFFFFFFF
```

4. 確認每個路徑的控制器狀態均為有效、且具有正確的ANA狀態：

```
nvme list-subsys /dev/nvme0n1
```

展示 **NVMe/FC** 範例

```
nvme-subsys0 - NQN=nqn.1992-
08.com.netapp:sn.0cd9ee0dc0ec11ee8e7fd039ea9e8ae9:subsystem.nvme
\
 +- nvme1 fc traddr=nn-0x2005d039eaa7dfc8:pn-0x2086d039eaa7dfc8
host_traddr=nn-0x20000024ff752e6d:pn-0x21000024ff752e6d live non-
optimized
 +- nvme2 fc traddr=nn-0x2005d039eaa7dfc8:pn-0x2016d039eaa7dfc8
host_traddr=nn-0x20000024ff752e6c:pn-0x21000024ff752e6c live
optimized
 +- nvme3 fc traddr=nn-0x2005d039eaa7dfc8:pn-0x2081d039eaa7dfc8
host_traddr=nn-0x20000024ff752e6c:pn-0x21000024ff752e6c live non-
optimized
 +- nvme4 fc traddr=nn-0x2005d039eaa7dfc8:pn-0x2087d039eaa7dfc8
host_traddr=nn-0x20000024ff752e6d:pn-0x21000024ff752e6d live
optimized
```

展示 **NVMe/TCP** 範例

```
nvme-subsys0 - NQN=nqn.1992-
08.com.netapp:sn.a1b2b785b9de11ee8e7fd039ea9e8ae9:subsystem.nvme_tcp
_1
\
 +- nvme0 tcp traddr=192.168.2.26 trsvcid=4420
host_traddr=192.168.2.31 live non-optimized
 +- nvme1 tcp traddr=192.168.2.25 trsvcid=4420
host_traddr=192.168.2.31 live optimized
 +- nvme2 tcp traddr=192.168.1.25 trsvcid=4420
host_traddr=192.168.1.31 live non-optimized
 +- nvme3 tcp traddr=192.168.1.24 trsvcid=4420
host_traddr=192.168.1.31 live optimized
```

5. 確認已在主機上建立並正確探索命名空間：

```
nvme list
```

顯示範例

```
Node         SN                   Model
------------------------------------------------------------
/dev/nvme4n1 81Ix2BVuekWcAAAAAAAB   NetApp ONTAP Controller


Namespace Usage    Format             FW             Rev
-----------------------------------------------------------------
1                 21.47 GB / 21.47 GB   4 KiB + 0 B   FFFFFFFF
```

6. 驗證NetApp外掛程式是否顯示每ONTAP 個版本名稱空間裝置的正確值：

欄位

```
nvme netapp ontapdevices -o column
```

顯示範例

```
Device        Vserver       Namespace Path
-----------------------------------------------------
/dev/nvme0n1  tcpiscsi_129  /vol/tcpnvme_1_0_0/tcpnvme_ns



NSID       UUID                                   Size
-----------------------------------------------------------------
1          05c2c351-5d7f-41d7-9bd8-1a56c            21.47GB
```

JSON

```
nvme netapp ontapdevices -o json
```

顯示範例

```
{
  "ONTAPdevices": [
    {
      "Device": "/dev/nvme0n1",
      "Vserver": "tcpiscsi_129",
      "Namespace Path”: /vol/tcpnvme_1_0_0/tcpnvme_ns ",
      "NSID": 1,
      "UUID": " 05c2c351-5d7f-41d7-9bd8-1a56c160c80b ",
      "Size2: "21.47GB",
      "LBA_Data_Size": 4096,
      "Namespace Size" : 5242880
    },
  ]
}
```

步驟 6 ：檢閱已知問題

這些是已知問題：

NetApp錯誤ID	標題	說明
"1479047"	RHEL 8.x NVMe-oF 主機建立重複的持久發現控制器 (PDC)	在 NVMe-oF 主機上，可以使用「nvme discover -p」指令建立 PDC。使用此指令時，每個發起方-目標組合只能建立一個 PDC。但是，如果您在 NVMe-oF 主機上執行 RHEL 8.x，則每次執行「nvme discover -p」時都會建立重複的 PDC。這會導致主機和目標裝置上的資源被不必要地消耗掉。

版權資訊

Copyright © 2026 NetApp, Inc. 版權所有。台灣印製。非經版權所有人事先書面同意，不得將本受版權保護文件的任何部分以任何形式或任何方法（圖形、電子或機械）重製，包括影印、錄影、錄音或儲存至電子檢索系統中。

由 NetApp 版權資料衍伸之軟體必須遵守下列授權和免責聲明：

此軟體以 NETAPP「原樣」提供，不含任何明示或暗示的擔保，包括但不限於有關適售性或特定目的適用性之擔保，特此聲明。於任何情況下，就任何已造成或基於任何理論上責任之直接性、間接性、附隨性、特殊性、懲罰性或衍生性損害（包括但不限於替代商品或服務之採購；使用、資料或利潤上的損失；或企業營運中斷），無論是在使用此軟體時以任何方式所產生的契約、嚴格責任或侵權行為（包括疏忽或其他）等方面，NetApp 概不負責，即使已被告知有前述損害存在之可能性亦然。

NetApp 保留隨時變更本文所述之任何產品的權利，恕不另行通知。NetApp 不承擔因使用本文所述之產品而產生的責任或義務，除非明確經過 NetApp 書面同意。使用或購買此產品並不會在依據任何專利權、商標權或任何其他 NetApp 智慧財產權的情況下轉讓授權。

本手冊所述之產品受到一項（含）以上的美國專利、國外專利或申請中專利所保障。

有限權利說明：政府機關的使用、複製或公開揭露須受 DFARS 252.227-7013（2014 年 2 月）和 FAR 52.227-19（2007 年 12 月）中的「技術資料權利 - 非商業項目」條款 (b)(3) 小段所述之限制。

此處所含屬於商業產品和 / 或商業服務（如 FAR 2.101 所定義）的資料均為 NetApp, Inc. 所有。根據本協議提供的所有 NetApp 技術資料和電腦軟體皆屬於商業性質，並且完全由私人出資開發。 美國政府對於該資料具有非專屬、非轉讓、非轉授權、全球性、有限且不可撤銷的使用權限，僅限於美國政府為傳輸此資料所訂合約所允許之範圍，並基於履行該合約之目的方可使用。除非本文另有規定，否則未經 NetApp Inc. 事前書面許可，不得逕行使用、揭露、重製、修改、履行或展示該資料。美國政府授予國防部之許可權利，僅適用於 DFARS 條款 252.227-7015(b)（2014 年 2 月）所述權利。

商標資訊

NETAPP、NETAPP 標誌及 http://www.netapp.com/TM 所列之標章均為 NetApp, Inc. 的商標。文中所涉及的所有其他公司或產品名稱，均為其各自所有者的商標，不得侵犯。