



Rocky Linux

ONTAP SAN Host Utilities

NetApp
January 30, 2026

This PDF was generated from <https://docs.netapp.com/it-it/ontap-sanhost/nvme-rockylinux-supported-features.html> on January 30, 2026. Always check docs.netapp.com for the latest.

Sommario

Rocky Linux	1
Scopri il supporto e le funzionalità ONTAP per Rocky Linux	1
Cosa c'è dopo?	1
Configurare Rocky Linux 10.x per NVMe-oF con storage ONTAP	2
Passaggio 1: Se lo si desidera, attivare l'avvio SAN	2
Passaggio 2: installa Rocky Linux e il software NVMe e verifica la configurazione	2
Passaggio 3: configurare NVMe/FC e NVMe/TCP	4
Passaggio 4: facoltativamente, abilitare 1 MB I/O per NVMe/FC	12
Passaggio 5: verificare i servizi di avvio NVMe	13
Passaggio 6: verificare la configurazione multipathing	14
Passaggio 7: impostare l'autenticazione in-band sicura	18
Fase 8: Esaminare i problemi noti	25
Configurare Rocky Linux 9.x per NVMe-oF con storage ONTAP	25
Passaggio 1: Se lo si desidera, attivare l'avvio SAN	25
Passaggio 2: installa Rocky Linux e il software NVMe e verifica la configurazione	25
Passaggio 3: configurare NVMe/FC e NVMe/TCP	27
Passaggio 4: facoltativamente, abilitare 1 MB I/O per NVMe/FC	39
Passaggio 5: verificare i servizi di avvio NVMe	40
Passaggio 6: verificare la configurazione multipathing	41
Passaggio 7: impostare l'autenticazione in-band sicura	45
Fase 8: Esaminare i problemi noti	53
Configurare Rocky Linux 8.x per NVMe-oF con storage ONTAP	53
Passaggio 1: installa Rocky Linux e il software NVMe e verifica la configurazione	53
Passaggio 2: configurare NVMe/FC e NVMe/TCP	55
Passaggio 3: Facoltativamente, abilitare 1 MB di I/O per NVMe/FC	66
Passaggio 4: verificare la configurazione del multipathing	67
Passaggio 5: rivedere i problemi noti	71

Rocky Linux

Scopri il supporto e le funzionalità ONTAP per Rocky Linux

Le funzionalità supportate per la configurazione host con NVMe over Fabrics (NVMe-oF) variano in base alla versione di ONTAP e Rocky Linux.

Caratteristica	Versione host Rocky Linux	Versione ONTAP
L'autenticazione in-band sicura è supportata tramite NVMe/TCP tra un host RHEL e un controller ONTAP	9.3 o successivo	9.12.1 o successivo
NVMe/TCP fornisce spazi dei nomi utilizzando il nativo <code>nvme-cli</code> pacchetto	8.2 o successivo	9.10.1 o successivo
NVMe/TCP è una funzionalità aziendale completamente supportata	9.0 o successivo	9.10.1 o successivo
Il traffico NVMe e SCSI è supportato sullo stesso host utilizzando NVMe multipath per gli spazi dei nomi NVMe-oF e dm-multipath per le LUN SCSI	8.2 o successivo	9.4 o successivo

ONTAP supporta le seguenti funzionalità host SAN indipendentemente dalla versione ONTAP in esecuzione sulla configurazione del sistema.

Caratteristica	Versione host Rocky Linux
Il multipathing NVMe nativo è abilitato per impostazione predefinita	10.0 o successivo
L'avvio SAN è abilitato tramite il protocollo NVMe/FC	9.4 o successivo
IL <code>nvme-cli</code> il pacchetto include script di connessione automatica che eliminano la necessità di script di terze parti	8.2 o successivo
La regola udev nativa nel <code>nvme-cli</code> pacchetto fornisce bilanciamento del carico round-robin per il multipathing NVMe	8.2 o successivo



Per i dettagli sulle configurazioni supportate, vedere ["Tool di matrice di interoperabilità"](#).

Cosa c'è dopo?

Se la tua versione di Rocky Linux è ..	Scopri di più su ..
10 serie	"Configurazione di NVMe per Rocky Linux 10.x"
Serie 9	"Configurazione di NVMe per Rocky Linux 9.x"
Serie 8	"Configurazione di NVMe per Rocky Linux 8.x"

Informazioni correlate

["Scopri di più sulla gestione dei protocolli NVMe"](#)

Configurare Rocky Linux 10.x per NVMe-oF con storage ONTAP

Gli host Rocky Linux supportano i protocolli NVMe su Fibre Channel (NVMe/FC) e NVMe su TCP (NVMe/TCP) con Asymmetric Namespace Access (ANA). ANA fornisce funzionalità multipathing equivalenti all'accesso asimmetrico alle unità logiche (ALUA) negli ambienti iSCSI e FCP.

Scopri come configurare gli host NVMe over Fabrics (NVMe-oF) per Rocky Linux 10.x. Per ulteriori informazioni sul supporto e sulle funzionalità, vedere ["Supporto e funzionalità di Rocky Linux ONTAP"](#).

NVMe-oF con Rocky Linux 10.x presenta le seguenti limitazioni note:

- Il `nvme disconnect-all` Il comando disconnette sia il file system root che quello dati e potrebbe causare instabilità del sistema. Non eseguire questa operazione su sistemi che si avviano da SAN tramite namespace NVMe-TCP o NVMe-FC.

Passaggio 1: Se lo si desidera, attivare l'avvio SAN

È possibile configurare l'host per utilizzare l'avvio SAN per semplificare la distribuzione e migliorare la scalabilità. Utilizzare il ["Tool di matrice di interoperabilità"](#) per verificare che il sistema operativo Linux, l'adattatore bus host (HBA), il firmware HBA, il BIOS di avvio HBA e la versione ONTAP supportino l'avvio SAN.

Fasi

1. ["Crea uno spazio dei nomi NVMe e mappalo all'host"](#) .
2. Abilitare l'avvio SAN nel BIOS del server per le porte su cui è mappato lo spazio dei nomi di avvio SAN.

Per informazioni su come attivare il BIOS HBA, consultare la documentazione specifica del vendor.

3. Riavviare l'host e verificare che il sistema operativo sia attivo e funzionante.

Passaggio 2: installa Rocky Linux e il software NVMe e verifica la configurazione

Per configurare l'host per NVMe-oF è necessario installare i pacchetti software host e NVMe, abilitare il multipathing e verificare la configurazione NQN dell'host.

Fasi

1. Installare Rocky Linux 10.x sul server. Una volta completata l'installazione, verifica di avere installato il kernel Rocky Linux 10.x richiesto:

```
uname -r
```

Esempio di versione del kernel Rocky Linux:

```
6.12.0-55.9.1.el10_0.x86_64
```

2. Installare `nvme-cli` pacchetto:

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

L'esempio seguente mostra un `nvme-cli` versione del pacchetto:

```
nvme-cli-2.11-5.el10.x86_64
```

3. Installare `libnvme` pacchetto:

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

L'esempio seguente mostra un `libnvme` versione del pacchetto:

```
libnvme-1.11.1-1.el10.x86_64
```

4. Sull'host, controlla la stringa `hostnqn` in `/etc/nvme/hostnqn`:

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

L'esempio seguente mostra un `hostnqn` valore:

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

5. Sul sistema ONTAP , verificare che `hostnqn` la stringa corrisponde a `hostnqn` stringa per il sottosistema corrispondente nell'array ONTAP :

```
::> vserver nvme subsystem host show -vserver vs_nvme_194_rockylinux10
```

Mostra esempio

```
Vserver Subsystem Priority Host NQN
-----
vs_nvme_194_rockylinux10
    nvme4
        regular    nqn.2014-
08.org.nvmexpress:uuid:4c4c4544-0056-5410-8048-c7c04f425633
    nvme_1
        regular    nqn.2014-
08.org.nvmexpress:uuid:4c4c4544-0056-5410-8048-c7c04f425633
    nvme_2
        regular    nqn.2014-
08.org.nvmexpress:uuid:4c4c4544-0056-5410-8048-c7c04f425633
    nvme_3
        regular    nqn.2014-
08.org.nvmexpress:uuid:4c4c4544-0056-5410-8048-c7c04f425633
4 entries were displayed.
```



Se il `hostnqn` le stringhe non corrispondono, utilizzare il `vserver modify` comando per aggiornare il `hostnqn` stringa sul sottosistema del sistema di archiviazione ONTAP corrispondente per abbinare `hostnqn` stringa da `/etc/nvme/hostnqn` sull'ospite.

Passaggio 3: configurare NVMe/FC e NVMe/TCP

Configurare NVMe/FC con adattatori Broadcom/Emulex o Marvell/QLogic oppure configurare NVMe/TCP utilizzando operazioni di rilevamento e connessione manuali.

NVMe/FC - Broadcom/Emulex

Configurare NVMe/FC per un adattatore Broadcom/Emulex.

Fasi

1. Verificare di utilizzare il modello di adattatore supportato:

a. Visualizza i nomi dei modelli:

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

Viene visualizzato il seguente output:

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

b. Visualizza le descrizioni dei modelli:

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

Dovresti vedere un output simile al seguente esempio:

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

2. Verificare di utilizzare il Broadcom consigliato lpfc firmware e driver della posta in arrivo:

a. Visualizza la versione del firmware:

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

Il comando restituisce le versioni del firmware:

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

b. Visualizza la versione del driver in arrivo:

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

L'esempio seguente mostra la versione del driver:

```
0:14.4.0.6
```

Per l'elenco aggiornato dei driver della scheda di rete supportati e delle versioni del firmware, vedere ["Tool di matrice di interoperabilità"](#).

3. Verificare che `lpfc_enable_fc4_type` è impostato su 3:

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

4. Verificare che sia possibile visualizzare le porte dell'iniziatore:

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

Dovresti vedere un output simile a:

```
0x2100f4c7aa0cd7c2  
0x2100f4c7aa0cd7c3
```

5. Verificare che le porte dell'iniziatore siano in linea:

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

Viene visualizzato il seguente output:

```
Online  
Online
```

6. Verificare che le porte iniziatore NVMe/FC siano abilitate e che le porte di destinazione siano visibili:

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

Mostra esempio

```
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 x202fd039eaa7dfc8 WWNN x202cd039eaa7dfc8
DID x021310 TARGET DISCSRVC ONLINE
NVME RPORT          WWPN x202dd039eaa7dfc8 WWNN x202cd039eaa7dfc8
DID x020b10 TARGET DISCSRVC ONLINE
```

```
NVME Statistics
LS: Xmt 0000000810 Cmpl 0000000810 Abort 00000000
LS XMIT: Err 00000000 CMPL: xb 00000000 Err 00000000
Total FCP Cmpl 000000007b098f07 Issue 000000007aee27c4 OutIO
ffffffffffffe498bd
          abort 000013b4 noxri 00000000 nondlp 00000058 qdepth
00000000 wqerr 00000000 err 00000000
FCP CMPL: xb 000013b4 Err 00021443
```

```
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 x2033d039eaa7dfc8 WWNN x202cd039eaa7dfc8
DID x020110 TARGET DISCSRVC ONLINE
NVME RPORT          WWPN x2032d039eaa7dfc8 WWNN x202cd039eaa7dfc8
DID x022910 TARGET DISCSRVC ONLINE
```

```
NVME Statistics
LS: Xmt 0000000840 Cmpl 0000000840 Abort 00000000
LS XMIT: Err 00000000 CMPL: xb 00000000 Err 00000000
Total FCP Cmpl 000000007afd4434 Issue 000000007ae31b83 OutIO
ffffffffffffe5d74f
          abort 000014a5 noxri 00000000 nondlp 0000006a qdepth
00000000 wqerr 00000000 err 00000000
FCP CMPL: xb 000014a5 Err 0002149a
```

NVMe/FC - Marvell/QLogic

Configurare NVMe/FC per un adattatore Marvell/QLogic.

Fasi

1. Verificare di utilizzare le versioni supportate del driver e del firmware dell'adattatore:

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

L'esempio seguente mostra le versioni del driver e del firmware:

```
QLE2872 FW:v9.15.00 DVR:v10.02.09.300-k  
QLE2872 FW:v9.15.00 DVR:v10.02.09.300-k
```

2. Verificare che `ql2xnvmeenable` è impostato. Ciò consente all'adattatore Marvell di funzionare come iniziatore NVMe/FC:

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

L'uscita prevista è 1.

NVMe/TCP

Il protocollo NVMe/TCP non supporta l'operazione di connessione automatica. In alternativa, è possibile scoprire i sottosistemi e gli spazi dei nomi NVMe/TCP eseguendo l'`NVMe/TCP connect O connect-all` operazioni manualmente.

Fasi

1. Verificare che la porta di avvio possa ottenere i dati della pagina del registro di individuazione attraverso i LIF NVMe/TCP supportati:

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

Mostra esempio

```
nvme discover -t tcp -w 192.168.20.1 -a 192.168.20.20

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: 4
trsvcid: 8009
subnqn: nqn.1992-
08.com.netapp:sn.64e65e6caae711ef9668d039ea951c46:discovery
traddr: 192.168.21.21
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: 2
trsvcid: 8009
subnqn: nqn.1992-
08.com.netapp:sn.64e65e6caae711ef9668d039ea951c46:discovery
traddr: 192.168.20.21
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: 3
trsvcid: 8009
subnqn: nqn.1992-
08.com.netapp:sn.64e65e6caae711ef9668d039ea951c46:discovery
traddr: 192.168.21.20
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:  1
trsvcid: 8009
subnqn:  nqn.1992-
08.com.netapp:sn.64e65e6caae711ef9668d039ea951c46:discovery
traddr:  192.168.20.20
eflags:  explicit discovery connections, duplicate discovery
information
sectype: none
=====Discovery Log Entry 4=====
trtype:  tcp
adrfam:  ipv4
subtype: nvme subsystem
treq:    not specified
portid:  4
trsvcid: 4420
subnqn:  nqn.1992-
08.com.netapp:sn.64e65e6caae711ef9668d039ea951c46:subsystem.rock
ylinux10_tcp_subsystem
traddr:  192.168.21.21
eflags:  none
sectype: none
=====Discovery Log Entry 5=====
trtype:  tcp
adrfam:  ipv4
subtype: nvme subsystem
treq:    not specified
portid:  2
trsvcid: 4420
subnqn:  nqn.1992-
08.com.netapp:sn.64e65e6caae711ef9668d039ea951c46:subsystem.rock
ylinux10_tcp_subsystem
traddr:  192.168.20.21
eflags:  none
sectype: none
=====Discovery Log Entry 6=====
trtype:  tcp
adrfam:  ipv4
subtype: nvme subsystem
treq:    not specified
portid:  3
trsvcid: 4420
subnqn:  nqn.1992-
08.com.netapp:sn.64e65e6caae711ef9668d039ea951c46:subsystem.rock

```

```

ylinux10_tcp_subsystem
traddr: 192.168.21.20
eflags: none
sectype: none
=====Discovery Log Entry 7=====
trtype: tcp
adrfam: ipv4
subtype: nvme subsystem
treq: not specified
portid: 1
trsvcid: 4420
subnqn: nqn.1992-
08.com.netapp:sn.64e65e6caae711ef9668d039ea951c46:subsystem.rock
ylinux10_tcp_subsystem
traddr: 192.168.20.20
eflags: none
sectype: none

```

2. Verificare che le altre combinazioni LIF NVMe/TCP initiator-target riescano a recuperare correttamente i dati della pagina del registro di individuazione:

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

Mostra esempio

```

nvme discover -t tcp -w 192.168.20.1 -a 192.168.20.20
nvme discover -t tcp -w 192.168.21.1 -a 192.168.21.20
nvme discover -t tcp -w 192.168.20.1 -a 192.168.20.21
nvme discover -t tcp -w 192.168.21.1 -a 192.168.21.21

```

3. Eseguire `nvme connect-all` Command tra tutti i LIF target initiator NVMe/TCP supportati nei nodi:

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

Mostra esempio

```
nvme      connect-all -t tcp -w 192.168.20.1 -a
192.168.20.20
nvme      connect-all -t tcp -w 192.168.21.1 -a
192.168.21.20
nvme      connect-all -t tcp -w 192.168.20.1 -a
192.168.20.21
nvme      connect-all -t tcp -w 192.168.21.1 -a
192.168.21.21
```

A partire da Rocky Linux 9.4, l'impostazione per NVMe/TCP `ctrl_loss_tmo` timeout viene automaticamente impostato su "off". Di conseguenza:

- Non ci sono limiti al numero di tentativi (tentativi illimitati).
- Non è necessario configurare manualmente uno specifico `ctrl_loss_tmo` timeout durata quando si utilizza il `nvme connect` o `nvme connect-all` comandi (opzione `-l`).
- I controller NVMe/TCP non subiscono timeout in caso di errore del percorso e rimangono connessi indefinitamente.

Passaggio 4: facoltativamente, abilitare 1 MB I/O per NVMe/FC

ONTAP segnala una dimensione massima di trasferimento dati (MDTS) pari a 8 nei dati Identify Controller. Ciò significa che la dimensione massima della richiesta di I/O può arrivare fino a 1 MB. Per emettere richieste di I/O di dimensione 1 MB per un host Broadcom NVMe/FC, è necessario aumentare il `lpfc` valore del `lpfc_sg_seg_cnt` parametro a 256 dal valore predefinito di 64.



Questi passaggi non si applicano agli host Qlogic NVMe/FC.

Fasi

1. Impostare il `lpfc_sg_seg_cnt` parametro su 256:

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

Dovresti vedere un output simile al seguente esempio:

```
options lpfc lpfc_sg_seg_cnt=256
```

2. Eseguire il `dracut -f` comando e riavviare l'host.
3. Verificare che il valore per `lpfc_sg_seg_cnt` sia 256:

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

Passaggio 5: verificare i servizi di avvio NVMe

IL `nvme-fc-boot-connections.service` E `nvme-f-autoconnect.service` servizi di avvio inclusi in NVMe/FC `nvme-cli` i pacchetti vengono abilitati automaticamente all'avvio del sistema.

Dopo aver completato l'avvio, verificare che `nvme-fc-boot-connections.service` E `nvme-f-autoconnect.service` i servizi di avvio sono abilitati.

Fasi

1. Verificare che `nvme-f-autoconnect.service` sia attivato:

```
systemctl status nvme-f-autoconnect.service
```

Mostra output di esempio

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

Jun 10 04:06:26 SR630-13-201.lab.eng.btc.netapp.in systemd[1]:
Starting Connect NVMe-oF subsystems automatically during boot...
Jun 10 04:06:26 SR630-13-201.lab.eng.btc.netapp.in systemd[1]: nvme-f-
autoconnect.service: Deactivated successfully.
Jun 10 04:06:26 SR630-13-201.lab.eng.btc.netapp.in systemd[1]:
Finished Connect NVMe-oF subsystems automatically during boot.
```

2. Verificare che `nvme-fc-boot-connections.service` sia attivato:

```
systemctl status nvme-fc-boot-connections.service
```

Mostra output di esempio

```
nvme-fc-boot-connections.service - Auto-connect to subsystems on FC-
NVME devices found during boot
   Loaded: loaded (/usr/lib/systemd/system/nvme-fc-boot-
connections.service; enabled; preset: enabled)
   Active: inactive (dead) since Tue 2025-06-10 01:08:36 EDT; 2h
59min ago
     Main PID: 7090 (code=exited, status=0/SUCCESS)
        CPU: 30ms

Jun 10 01:08:36 localhost systemd[1]: Starting Auto-connect to
subsystems on FC-NVME devices found during boot...
Jun 10 01:08:36 localhost systemd[1]: nvme-fc-boot-
connections.service: Deactivated successfully.
Jun 10 01:08:36 localhost systemd[1]: Finished Auto-connect to
subsystems on FC-NVME devices found during boot.
```

Passaggio 6: verificare la configurazione multipathing

Verificare che lo stato multipath NVMe in-kernel, lo stato ANA e i namespace ONTAP siano corretti per la configurazione NVMe-of.

Fasi

1. Verificare che il multipath NVMe nel kernel sia attivato:

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

Viene visualizzato il seguente output:

```
Y
```

2. Verificare che le impostazioni NVMe-of appropriate (ad esempio, modello impostato su controller NetApp ONTAP e ipolicy per il bilanciamento del carico impostato su round-robin) per i rispettivi spazi dei nomi ONTAP si riflettano correttamente sull'host:

- a. Visualizza i sottosistemi:

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

Viene visualizzato il seguente output:

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

b. Visualizza la politica:

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

Viene visualizzato il seguente output:

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

3. Verificare che gli spazi dei nomi siano stati creati e rilevati correttamente sull'host:

```
nvme list
```

Mostra esempio

Node	SN	Model	

/dev/nvme4n1	81Ix2BVuekWcAAAAAAB	NetApp ONTAP Controller	
Namespace Usage	Format	FW	Rev

1	21.47 GB / 21.47 GB	4 KiB + 0 B	FFFFFFFF

4. Verificare che lo stato del controller di ciascun percorso sia attivo e che abbia lo stato ANA corretto:

NVMe/FC

```
nvme list-subsys /dev/nvme5n1
```

Mostra esempio

```
nvme-subsys5 - NQN=nqn.1992-08.com.netapp:sn.f7565b15a66911ef9668d039ea951c46:subsystem.nvme
1
                    hostnqn=nqn.2014-08.org.nvmexpress:uuid:4c4c4544-0056-5410-8048-c7c04f425633
\
+- nvme126 fc traddr=nn-0x2036d039ea951c45:pn-0x2038d039ea951c45,host_traddr=nn-0x2000f4c7aa0cd7c3:pn-0x2100f4c7aa0cd7c3 live optimized
+- nvme176 fc traddr=nn-0x2036d039ea951c45:pn-0x2037d039ea951c45,host_traddr=nn-0x2000f4c7aa0cd7c2:pn-0x2100f4c7aa0cd7c2 live optimized
+- nvme5 fc traddr=nn-0x2036d039ea951c45:pn-0x2039d039ea951c45,host_traddr=nn-0x2000f4c7aa0cd7c2:pn-0x2100f4c7aa0cd7c2 live non-optimized
+- nvme71 fc traddr=nn-0x2036d039ea951c45:pn-0x203ad039ea951c45,host_traddr=nn-0x2000f4c7aa0cd7c3:pn-0x2100f4c7aa0cd7c3 live non-optimized
```

NVMe/TCP

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

Mostra esempio

```
nvme-subsys4 - NQN=nqn.1992-08.com.netapp:sn.64e65e6caae711ef9668d039ea951c46:subsystem.nvme4
                hostnqn=nqn.2014-08.org.nvmexpress:uuid:4c4c4544-0035-5910-804b-c2c04f444d33
\
+- nvme102 tcp
traddr=192.168.21.20,trsvcid=4420,host_traddr=192.168.21.1,src_addr=192.168.21.1 live non-optimized
+- nvme151 tcp
traddr=192.168.21.21,trsvcid=4420,host_traddr=192.168.21.1,src_addr=192.168.21.1 live optimized
+- nvme4 tcp
traddr=192.168.20.20,trsvcid=4420,host_traddr=192.168.20.1,src_addr=192.168.20.1 live non-optimized
+- nvme53 tcp
traddr=192.168.20.21,trsvcid=4420,host_traddr=192.168.20.1,src_addr=192.168.20.1 live optimized
```

5. Verificare che il plug-in NetApp visualizzi i valori corretti per ciascun dispositivo dello spazio dei nomi ONTAP:

Colonna

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

Mostra esempio

Device	Vserver	Namespace	Path
/dev/nvme10n1	vs_tcp_rockylinux10		/vol/vol10/ns10

NSID	UUID	Size
1	bbf51146-fc64-4197-b8cf-8a24f6f359b3	21.47GB

JSON

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

Mostra esempio

```
{
  "ONTAPdevices":[
    {
      "Device":"/dev/nvme10n1",
      "Vserver":"vs_tcp_rockylinux10",
      "Namespace_Path":"/vol/vol10/ns10",
      "NSID":1,
      "UUID":"bbf51146-fc64-4197-b8cf-8a24f6f359b3",
      "Size":"21.47GB",
      "LBA_Data_Size":4096,
      "Namespace_Size":5242880
    }
  ]
}
```

Passaggio 7: impostare l'autenticazione in-band sicura

L'autenticazione in-band sicura è supportata tramite NVMe/TCP tra un host Rocky Linux 10.x e un controller ONTAP .

Ogni host o controller deve essere associato a un `DH-HMAC-CHAP` chiave per impostare l'autenticazione sicura. Una `DH-HMAC-CHAP` La chiave è una combinazione dell'`NQN` dell'host o del controller NVMe e di un segreto di autenticazione configurato dall'amministratore. Per autenticare il proprio peer, un host o un controller NVMe deve riconoscere la chiave associata al peer.

Fasi

Imposta l'autenticazione in-band sicura tramite la CLI o un file JSON di configurazione. Se è necessario specificare chiavi `dhchap` diverse per sottosistemi diversi, è necessario utilizzare un file di configurazione JSON.

CLI

Configurare l'autenticazione in banda protetta utilizzando la CLI.

1. Ottenere l'NQN dell'host:

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

2. Genera la chiave dhchap per l'host Rocky Linux 10.x.

L'output seguente descrive il `gen-dhchap-key` parametri del comando:

```
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

Nell'esempio seguente, viene generata una chiave casuale dhCHAP con HMAC impostato su 3 (SHA-512).

```
nvme gen-dhchap-key -m 3 -n nqn.2014-
08.org.nvmexpress:uuid:4c4c4544-0035-5910-804b-c2c04f444d33
DHHC-
1:03:7zf8I9gaRcDWH3tCH5vLgaoyjzPIvwNWusBfKdpJa+hialaKDKJQ2o53pX3wYM9
xdv5DtKNNhJInZ7X8wU2RQpQIngc=:
```

3. Sul controller ONTAP, aggiungere l'host e specificare entrambe le chiavi dhchap:

```
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. Un host supporta due tipi di metodi di autenticazione, unidirezionale e bidirezionale. Sull'host, connettersi al controller ONTAP e specificare le chiavi dhchap in base al metodo di autenticazione scelto:

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

5. Convalidare `nvme connect authentication` comando verificando le chiavi dhchap dell'host e del controller:

a. Verificare le chiavi dhchap dell'host:

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

Mostra output di esempio per una configurazione unidirezionale

```
cat /sys/class/nvme-subsystem/nvme-subsys1/nvme*/dhchap_secret  
DHHC-  
1:03:fMCRJharXUOqRoIsOEaG6m2PH1yYvu5+z3jTmzEKUbcWu26I33b93b  
i12WR09XDho/1d3L45J+0FeCsStBEAfhYgkQU=:  
DHHC-  
1:03:fMCRJharXUOqRoIsOEaG6m2PH1yYvu5+z3jTmzEKUbcWu26I33b93b  
i12WR09XDho/1d3L45J+0FeCsStBEAfhYgkQU=:  
DHHC-  
1:03:fMCRJharXUOqRoIsOEaG6m2PH1yYvu5+z3jTmzEKUbcWu26I33b93b  
i12WR09XDho/1d3L45J+0FeCsStBEAfhYgkQU=:  
DHHC-  
1:03:fMCRJharXUOqRoIsOEaG6m2PH1yYvu5+z3jTmzEKUbcWu26I33b93b  
i12WR09XDho/1d3L45J+0FeCsStBEAfhYgkQU=:
```

b. Verificare i tasti dhchap del controller:

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

Mostra output di esempio per una configurazione bidirezionale

```
cat /sys/class/nvme-subsystem/nvme-  
subsys6/nvme*/dhchap_ctrl_secret  
DHHC- 1:03:7zf8I9gaRcDWH3tCH5vLGaoyjzPIvwNWusBfKdpJa+hia  
1aKDKJQ2o53pX3wYM9xdv5DtKNNhJInZ7X8wU2RQpQIngc=:  
  
DHHC- 1:03:7zf8I9gaRcDWH3tCH5vLGaoyjzPIvwNWusBfKdpJa+hia  
1aKDKJQ2o53pX3wYM9xdv5DtKNNhJInZ7X8wU2RQpQIngc=:  
  
DHHC- 1:03:7zf8I9gaRcDWH3tCH5vLGaoyjzPIvwNWusBfKdpJa+hia  
1aKDKJQ2o53pX3wYM9xdv5DtKNNhJInZ7X8wU2RQpQIngc=:  
  
DHHC- 1:03:7zf8I9gaRcDWH3tCH5vLGaoyjzPIvwNWusBfKdpJa+hia  
1aKDKJQ2o53pX3wYM9xdv5DtKNNhJInZ7X8wU2RQpQIngc=:
```

JSON

Quando più sottosistemi NVMe sono disponibili sul controller ONTAP , è possibile utilizzare `/etc/nvme/config.json` file con il `nvme connect-all` comando.

Utilizzare il `-o` opzione per generare il file JSON. Per ulteriori opzioni di sintassi, fare riferimento alle pagine man di NVMe connect-all.

1. Configurare il file JSON.



Nell'esempio seguente, `dhchap_key` corrisponde a `dhchap_secret` E `dhchap_ctrl_key` corrisponde a `dhchap_ctrl_secret`.

Mostra esempio

```
cat /etc/nvme/config.json
[
  {
    "hostnqn": "nqn.2014-08.org.nvmexpress:uuid:4c4c4544-0035-
5910-804b-c2c04f444d33",
    "hostid": "4c4c4544-0035-5910-804b-c2c04f444d33",
    "dhchap_key": "DHHC-
1:03:7zf8I9gaRcDWH3tCH5vLGaoyjzPIvwNWusBfKdpJa+hialaKDKJQ2o53pX3
wYM9xdv5DtKNNhJInZ7X8wU2RQpQIngc=",
    "subsystems": [
      {
        "nqn": "nqn.1992-
08.com.netapp:sn.127ade26168811f0a50ed039eab69ad3:subsystem.inba
nd_unidirectional",
        "ports": [
          {
            "transport": "tcp",
            "traddr": "192.168.20.17",
            "host_traddr": "192.168.20.1",
            "trsvcid": "4420"
          },
          {
            "transport": "tcp",
            "traddr": "192.168.20.18",
            "host_traddr": "192.168.20.1",
            "trsvcid": "4420"
          },
          {
            "transport": "tcp",
            "traddr": "192.168.21.18",
            "host_traddr": "192.168.21.1",
            "trsvcid": "4420"
          },
          {
            "transport": "tcp",
            "traddr": "192.168.21.17",
            "host_traddr": "192.168.21.1",
            "trsvcid": "4420"
          }
        ]
      }
    ]
  }
]
```

2. Connettersi al controller ONTAP utilizzando il file di configurazione JSON:

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

Mostra esempio

```
traddr=192.168.20.20 is already connected
traddr=192.168.20.20 is already connected
traddr=192.168.20.20 is already connected
traddr=192.168.20.20 is already connected
traddr=192.168.20.20 is already connected
traddr=192.168.20.20 is already connected
traddr=192.168.20.20 is already connected
traddr=192.168.20.20 is already connected
traddr=192.168.20.21 is already connected
traddr=192.168.20.21 is already connected
traddr=192.168.20.21 is already connected
traddr=192.168.20.21 is already connected
traddr=192.168.20.21 is already connected
traddr=192.168.20.21 is already connected
traddr=192.168.20.21 is already connected
traddr=192.168.20.21 is already connected
```

3. Verificare che i segreti dhchap siano stati abilitati per i rispettivi controller per ciascun sottosistema.

a. Verificare le chiavi dhchap dell'host:

```
cat /sys/class/nvme-subsystem/nvme-subsys0/nvme0/dhchap_secret
```

L'esempio seguente mostra una chiave dhchap:

```
DHHC-1:03:7zf8I9gaRcDWH3tCH5vLGAoyjzPIvwNWusBfKdpJa+hial
aKDKJQ2o53pX3wYM9xdv5DtKNNhJInZ7X8wU2RQpQIngc=:
```

b. Verificare i tasti dhchap del controller:

```
cat /sys/class/nvme-subsystem/nvme-
subsys0/nvme0/dhchap_ctrl_secret
```

Dovresti vedere un output simile al seguente esempio:

```
DHHC-1:03:fMcrJharXUOqRoIsOEaG6m2PH1yYvu5+z3jT  
mzEKUbcWu26I33b93bi12WR09XDho/1d3L45J+0FeCsStBEAfhYgkQU=:
```

Fase 8: Esaminare i problemi noti

Non ci sono problemi noti.

Configurare Rocky Linux 9.x per NVMe-oF con storage ONTAP

Gli host Rocky Linux supportano i protocolli NVMe su Fibre Channel (NVMe/FC) e NVMe su TCP (NVMe/TCP) con Asymmetric Namespace Access (ANA). ANA fornisce funzionalità multipathing equivalenti all'accesso asimmetrico alle unità logiche (ALUA) negli ambienti iSCSI e FCP.

Scopri come configurare gli host NVMe over Fabrics (NVMe-oF) per Rocky Linux 9.x. Per ulteriori informazioni sul supporto e sulle funzionalità, vedere ["Supporto e funzionalità di Rocky Linux ONTAP"](#).

NVMe-oF con Rocky Linux 9.x presenta le seguenti limitazioni note:

- IL `nvme disconnect-all` Il comando disconnette sia il file system root che quello dati e potrebbe causare instabilità del sistema. Non eseguire questa operazione su sistemi che si avviano da SAN tramite namespace NVMe-TCP o NVMe-FC.

Passaggio 1: Se lo si desidera, attivare l'avvio SAN

È possibile configurare l'host per utilizzare l'avvio SAN per semplificare la distribuzione e migliorare la scalabilità. Utilizzare il ["Tool di matrice di interoperabilità"](#) per verificare che il sistema operativo Linux, l'adattatore bus host (HBA), il firmware HBA, il BIOS di avvio HBA e la versione ONTAP supportino l'avvio SAN.

Fasi

1. ["Crea uno spazio dei nomi NVMe e mappalo all'host"](#) .
2. Abilitare l'avvio SAN nel BIOS del server per le porte su cui è mappato lo spazio dei nomi di avvio SAN.

Per informazioni su come attivare il BIOS HBA, consultare la documentazione specifica del vendor.

3. Riavviare l'host e verificare che il sistema operativo sia attivo e funzionante.

Passaggio 2: installa Rocky Linux e il software NVMe e verifica la configurazione

Per configurare l'host per NVMe-oF è necessario installare i pacchetti software host e NVMe, abilitare il multipathing e verificare la configurazione NQN dell'host.

Fasi

1. Installare Rocky Linux 9.x sul server. Una volta completata l'installazione, verifica di avere installato il kernel Rocky Linux 9.x richiesto:

```
uname -r
```

Esempio di versione del kernel Rocky Linux:

```
5.14.0-570.12.1.el9_6.x86_64
```

2. Installare `nvme-cli` pacchetto:

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

L'esempio seguente mostra una versione del pacchetto `nvme-cli`:

```
nvme-cli-2.11-5.el9.x86_64
```

3. Installare `libnvme` pacchetto:

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

L'esempio seguente mostra un `libnvme` versione del pacchetto:

```
libnvme-1.11.1-1.el9.x86_64
```

4. Sull'host Rocky Linux, controlla la stringa `hostnqn` in `/etc/nvme/hostnqn` :

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

L'esempio seguente mostra un `hostnqn` versione:

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

5. Sul sistema ONTAP , verificare che `hostnqn` la stringa corrisponde a `hostnqn` stringa per il sottosistema corrispondente nell'array ONTAP :

```
::> vserver nvme subsystem host show -vserver vs_coexistence_LPE36002
```

Mostra esempio

```
Vserver Subsystem Priority Host NQN
-----
vs_coexistence_LPE36002
    nvme
        regular    nqn.2014-
08.org.nvmexpress:uuid:4c4c4544-0056-5410-8048-b9c04f425633
    nvme_1
        regular    nqn.2014-
08.org.nvmexpress:uuid:4c4c4544-0056-5410-8048-b9c04f425633
    nvme_2
        regular    nqn.2014-
08.org.nvmexpress:uuid:4c4c4544-0056-5410-8048-b9c04f425633
    nvme_3
        regular    nqn.2014-
08.org.nvmexpress:uuid:4c4c4544-0056-5410-8048-b9c04f425633
4 entries were displayed.
```



Se il `hostnqn` le stringhe non corrispondono, utilizzare `vserver modify` per aggiornare `hostnqn` Stringa sul sottosistema di array ONTAP corrispondente a `hostnqn` stringa da `/etc/nvme/hostnqn` sull'host.

Passaggio 3: configurare NVMe/FC e NVMe/TCP

Configurare NVMe/FC con adattatori Broadcom/Emulex o Marvell/QLogic oppure configurare NVMe/TCP utilizzando operazioni di rilevamento e connessione manuali.

NVMe/FC - Broadcom/Emulex

Configurare NVMe/FC per un adattatore Broadcom/Emulex.

Fasi

1. Verificare di utilizzare il modello di adattatore supportato:

a. Visualizza i nomi dei modelli:

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

Viene visualizzato il seguente output:

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

b. Visualizza le descrizioni dei modelli:

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

Dovresti vedere un output simile al seguente esempio:

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

2. Verificare di utilizzare il Broadcom consigliato lpfc firmware e driver della posta in arrivo:

a. Visualizza la versione del firmware:

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

Il comando restituisce le versioni del firmware:

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

b. Visualizza la versione del driver in arrivo:

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

L'esempio seguente mostra la versione del driver:

```
0:14.4.0.6
```

Per l'elenco aggiornato dei driver della scheda di rete supportati e delle versioni del firmware, vedere ["Tool di matrice di interoperabilità"](#).

3. Verificare che `lpfc_enable_fc4_type` è impostato su 3:

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

4. Verificare che sia possibile visualizzare le porte dell'iniziatore:

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

L'esempio seguente mostra le identità delle porte:

```
0x2100f4c7aa0cd7c2  
0x2100f4c7aa0cd7c3
```

5. Verificare che le porte dell'iniziatore siano in linea:

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

Viene visualizzato il seguente output:

```
Online  
Online
```

6. Verificare che le porte iniziatore NVMe/FC siano abilitate e che le porte di destinazione siano visibili:

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

Mostra esempio

```
NVME Initiator Enabled
XRI Dist lpfc0 Total 6144 IO 5894 ELS 250
NVME LPORT lpfc0 WWPN x100000109b954518 WWNN x200000109b954518
DID x000000 ONLINE
```

```
NVME Statistics
LS: Xmt 0000000000 Cmpl 0000000000 Abort 00000000
LS XMIT: Err 00000000 Cmpl: xb 00000000 Err 00000000
Total FCP Cmpl 0000000000000000 Issue 0000000000000000 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 Statistics
LS: Xmt 0000000000 Cmpl 0000000000 Abort 00000000
LS XMIT: Err 00000000 Cmpl: xb 00000000 Err 00000000
Total FCP Cmpl 0000000000000000 Issue 0000000000000000 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 lpfc2 Total 6144 IO 5894 ELS 250
NVME LPORT lpfc2 WWPN x100000109bf044b1 WWNN x200000109bf044b1
DID x022a00 ONLINE
NVME RPORT          WWPN x200bd039eaa7dfc8 WWNN x2008d039eaa7dfc8
DID x021319 TARGET DISCSRV ONLINE
NVME RPORT          WWPN x2155d039eaa7dfc8 WWNN x2154d039eaa7dfc8
DID x02130f TARGET DISCSRV ONLINE
NVME RPORT          WWPN x2001d039eaa7dfc8 WWNN x2000d039eaa7dfc8
DID x021310 TARGET DISCSRV ONLINE
NVME RPORT          WWPN x200dd039eaa7dfc8 WWNN x2008d039eaa7dfc8
DID x020b15 TARGET DISCSRV ONLINE
NVME RPORT          WWPN x2156d039eaa7dfc8 WWNN x2154d039eaa7dfc8
DID x020b0d TARGET DISCSRV ONLINE
NVME RPORT          WWPN x2003d039eaa7dfc8 WWNN x2000d039eaa7dfc8
DID x020b10 TARGET DISCSRV ONLINE
```

```

NVME Statistics
LS: Xmt 0000003049 Cmpl 0000003049 Abort 00000000
LS XMIT: Err 00000000  CML: xb 00000000 Err 00000000
Total FCP Cmpl 0000000018f9450b Issue 0000000018f5de57 OutIO
ffffffffffffc994c
        abort 000036d3 noxri 00000313 nondlp 00000c8d qdepth
000000000 wqerr 00000064 err 00000000
FCP CML: xb 000036d1 Err 000fef0f

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 x2062d039eaa7dfc8 WWNN x2008d039eaa7dfc8
DID x022915 TARGET DISCSRV ONLINE
NVME RPORT          WWPN x2157d039eaa7dfc8 WWNN x2154d039eaa7dfc8
DID x02290f TARGET DISCSRV ONLINE
NVME RPORT          WWPN x2002d039eaa7dfc8 WWNN x2000d039eaa7dfc8
DID x022910 TARGET DISCSRV ONLINE
NVME RPORT          WWPN x2065d039eaa7dfc8 WWNN x2008d039eaa7dfc8
DID x020119 TARGET DISCSRV ONLINE
NVME RPORT          WWPN x2158d039eaa7dfc8 WWNN x2154d039eaa7dfc8
DID x02010d TARGET DISCSRV ONLINE
NVME RPORT          WWPN x2004d039eaa7dfc8 WWNN x2000d039eaa7dfc8
DID x020110 TARGET DISCSRV ONLINE

NVME Statistics
LS: Xmt 0000002f2c Cmpl 0000002f2c Abort 00000000
LS XMIT: Err 00000000  CML: xb 00000000 Err 00000000
Total FCP Cmpl 000000001aaf3eb5 Issue 000000001aab4373 OutIO
ffffffffffffc04be
        abort 000035cc noxri 0000038c nondlp 000009e3 qdepth
000000000 wqerr 00000082 err 00000000
FCP CML: xb 000035cc Err 000fcfc0

```

NVMe/FC - Marvell/QLogic

Configurare NVMe/FC per un adattatore Marvell/QLogic.

Fasi

1. Verificare di utilizzare le versioni supportate del driver e del firmware dell'adattatore:

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

L'esempio seguente mostra le versioni del driver e del firmware:

```
QLE2872 FW:v9.15.00 DVR:v10.02.09.300-k  
QLE2872 FW:v9.15.00 DVR:v10.02.09.300-k
```

2. Verificare che `ql2xnvmeenable` è impostato. Ciò consente all'adattatore Marvell di funzionare come iniziatore NVMe/FC:

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

L'uscita prevista è 1.

NVMe/TCP

Il protocollo NVMe/TCP non supporta l'operazione di connessione automatica. È necessario eseguire manualmente le operazioni di connessione o di connessione completa NVMe/TCP per individuare i sottosistemi e gli spazi dei nomi NVMe/TCP.

Fasi

1. Verificare che la porta di avvio possa ottenere i dati della pagina del registro di individuazione attraverso i LIF NVMe/TCP supportati:

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

Mostra esempio

```
nvme discover -t tcp -w 192.168.1.31 -a 192.168.1.24

Discovery Log Number of Records 20, Generation counter 25
=====Discovery Log Entry 0=====
trtype:  tcp
adrfam:  ipv4
subtype: current discovery subsystem
treq:    not specified
portid:  4
trsvcid: 8009
subnqn:  nqn.1992-
08.com.netapp:sn.0f4bale74eb611ef9f50d039eab6cb6d:discovery
traddr:  192.168.2.25
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:  2
trsvcid: 8009
subnqn:  nqn.1992-
08.com.netapp:sn.0f4bale74eb611ef9f50d039eab6cb6d:discovery
traddr:  192.168.1.25
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:  5
trsvcid: 8009
subnqn:  nqn.1992-
08.com.netapp:sn.0f4bale74eb611ef9f50d039eab6cb6d:discovery
traddr:  192.168.2.24
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:  1
trsvcid: 8009
subnqn:  nqn.1992-
08.com.netapp:sn.0f4ba1e74eb611ef9f50d039eab6cb6d:discovery
traddr:  192.168.1.24
eflags:  explicit discovery connections, duplicate discovery
information
sectype: none
=====Discovery Log Entry 4=====
trtype:  tcp
adrfam:  ipv4
subtype: nvme subsystem
treq:    not specified
portid:  4
trsvcid: 4420
subnqn:  nqn.1992-
08.com.netapp:sn.0f4ba1e74eb611ef9f50d039eab6cb6d:subsystem.
Unidirectional_DHCP_NONE_1_3
traddr:  192.168.2.25
eflags:  none
sectype: none
=====Discovery Log Entry 5=====
trtype:  tcp
adrfam:  ipv4
subtype: nvme subsystem
treq:    not specified
portid:  2
trsvcid: 4420
subnqn:  nqn.1992-
08.com.netapp:sn.0f4ba1e74eb611ef9f50d039eab6cb6d:subsystem.
Unidirectional_DHCP_NONE_1_4
traddr:  192.168.1.25
eflags:  none
sectype: none
=====Discovery Log Entry 6=====
trtype:  tcp
adrfam:  ipv4
subtype: nvme subsystem
treq:    not specified
portid:  5
trsvcid: 4420
subnqn:  nqn.1992-
08.com.netapp:sn.0f4ba1e74eb611ef9f50d039eab6cb6d:subsystem.

```

```

Unidirectional_DHCP_NONE_1_5
traddr: 192.168.2.24
eflags: none
sectype: none
=====Discovery Log Entry 7=====
trtype: tcp
adrfam: ipv4
subtype: nvme subsystem
treq: not specified
portid: 1
trsvcid: 4420
subnqn: nqn.1992-
08.com.netapp:sn.0f4bale74eb611ef9f50d039eab6cb6d:subsystem.
Unidirectional_DHCP_2_2
traddr: 192.168.1.24
eflags: none
sectype: none
=====Discovery Log Entry 8=====
trtype: tcp
adrfam: ipv4
subtype: nvme subsystem
treq: not specified
portid: 4
trsvcid: 4420
subnqn: nqn.1992-
08.com.netapp:sn.0f4bale74eb611ef9f50d039eab6cb6d:subsystem.
Unidirectional_DHCP_2_3
traddr: 192.168.2.25
eflags: none
sectype: none
=====Discovery Log Entry 9=====
trtype: tcp
adrfam: ipv4
subtype: nvme subsystem
treq: not specified
portid: 2
trsvcid: 4420
subnqn: nqn.1992-
08.com.netapp:sn.0f4bale74eb611ef9f50d039eab6cb6d:subsystem.
Unidirectional_DHCP_2_5
traddr: 192.168.1.25
eflags: none
sectype: none
=====Discovery Log Entry 10=====
trtype: tcp
adrfam: ipv4

```

```

subtype: nvme subsystem
treq:    not specified
portid:  5
trsvcid: 4420
subnqn:  nqn.1992-
08.com.netapp:sn.0f4bale74eb611ef9f50d039eab6cb6d:subsystem.
Bidirectional_DHCP_2_2
traddr:  192.168.2.24
eflags:  none
sectype: none
=====Discovery Log Entry 11=====
trtype:  tcp
adrfam:  ipv4
subtype: nvme subsystem
treq:    not specified
portid:  1
trsvcid: 4420
subnqn:  nqn.1992-
08.com.netapp:sn.0f4bale74eb611ef9f50d039eab6cb6d:subsystem.
Bidirectional_DHCP_2_3
traddr:  192.168.1.24
eflags:  none
sectype: none
=====Discovery Log Entry 12=====
trtype:  tcp
adrfam:  ipv4
subtype: nvme subsystem
treq:    not specified
portid:  4
trsvcid: 4420
subnqn:  nqn.1992-
08.com.netapp:sn.0f4bale74eb611ef9f50d039eab6cb6d:subsystem.
Bidirectional_DHCP_2_3
traddr:  192.168.2.25
eflags:  none
sectype: none
=====Discovery Log Entry 13=====
trtype:  tcp
adrfam:  ipv4
subtype: nvme subsystem
treq:    not specified
portid:  2
trsvcid: 4420
subnqn:  nqn.1992-
08.com.netapp:sn.0f4bale74eb611ef9f50d039eab6cb6d:subsystem.
Bidirectional_DHCP_NONE_2_4

```

```

traddr: 192.168.1.25
eflags: none
sectype: none
=====Discovery Log Entry 14=====
trtype: tcp
adrfam: ipv4
subtype: nvme subsystem
treq: not specified
portid: 5
trsvcid: 4420
subnqn: nqn.1992-
08.com.netapp:sn.0f4bale74eb611ef9f50d039eab6cb6d:subsystem.
Bidirectional_DHCP_NONE_2_5
traddr: 192.168.2.24
eflags: none
sectype: none
=====Discovery Log Entry 15=====
trtype: tcp
adrfam: ipv4
subtype: nvme subsystem
treq: not specified
portid: 1
trsvcid: 4420
subnqn: nqn.1992-
08.com.netapp:sn.0f4bale74eb611ef9f50d039eab6cb6d:subsystem.
Bidirectional_DHCP_NONE_2_6
traddr: 192.168.1.24
eflags: none
sectype: none
=====Discovery Log Entry 16=====
trtype: tcp
adrfam: ipv4
subtype: nvme subsystem
treq: not specified
portid: 4
trsvcid: 4420
subnqn: nqn.1992-
08.com.netapp:sn.0f4bale74eb611ef9f50d039eab6cb6d:subsystem.
Bidirectional_DHCP_NONE_2_7
traddr: 192.168.2.25
eflags: none
sectype: none
=====Discovery Log Entry 17=====
trtype: tcp
adrfam: ipv4
subtype: nvme subsystem

```

```

treq:    not specified
portid:  2
trsvcid: 4420
subnqn:  nqn.1992-
08.com.netapp:sn.0f4ba1e74eb611ef9f50d039eab6cb6d:subsystem.
Bidirectional_DHCP_NONE_2_8
traddr:  192.168.1.25
eflags:  none
sectype: none
=====Discovery Log Entry 18=====
trtype:  tcp
adrfam:  ipv4
subtype: nvme subsystem
treq:    not specified
portid:  5
trsvcid: 4420
subnqn:  nqn.1992-
08.com.netapp:sn.0f4ba1e74eb611ef9f50d039eab6cb6d:subsystem.nvme
_tcp_2
traddr:  192.168.2.24
eflags:  none
sectype: none
=====Discovery Log Entry 19=====
trtype:  tcp
adrfam:  ipv4
subtype: nvme subsystem
treq:    not specified
portid:  1
trsvcid: 4420
subnqn:  nqn.1992-
08.com.netapp:sn.0f4ba1e74eb611ef9f50d039eab6cb6d:subsystem.
Bidirectional_DHCP_NONE_2_9
traddr:  192.168.1.24
eflags:  none
sectype: none

```

2. Verificare che le altre combinazioni LIF NVMe/TCP initiator-target riescano a ottenere correttamente i dati della pagina del registro di individuazione:

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

Mostra esempio

```
nvme discover -t tcp -w 192.168.1.31 -a 192.168.1.24
nvme discover -t tcp -w 192.168.2.31 -a 192.168.2.24
nvme discover -t tcp -w 192.168.1.31 -a 192.168.1.25
nvme discover -t tcp -w 192.168.2.31 -a 192.168.2.25
```

3. Eseguire `nvme connect-all` Command tra tutti i LIF target initiator NVMe/TCP supportati nei nodi:

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

Mostra esempio

```
nvme      connect-all -t tcp -w 192.168.1.31 -a 192.168.1.24
nvme      connect-all -t tcp -w 192.168.2.31 -a 192.168.2.24
nvme      connect-all -t tcp -w 192.168.1.31 -a 192.168.1.25
nvme      connect-all -t tcp -w 192.168.2.31 -a 192.168.2.25
```

A partire da Rocky Linux 9.4, l'impostazione per NVMe/TCP `ctrl_loss_tmo` timeout viene automaticamente impostato su "off". Di conseguenza:

- Non ci sono limiti al numero di tentativi (tentativi illimitati).
- Non è necessario configurare manualmente uno specifico `ctrl_loss_tmo` timeout durata quando si utilizza il `nvme connect` o `nvme connect-all` comandi (opzione `-l`).
- I controller NVMe/TCP non subiscono timeout in caso di errore del percorso e rimangono connessi indefinitamente.

Passaggio 4: facoltativamente, abilitare 1 MB I/O per NVMe/FC

ONTAP segnala una dimensione massima di trasferimento dati (MDTS) pari a 8 nei dati Identify Controller. Ciò significa che la dimensione massima della richiesta di I/O può arrivare fino a 1 MB. Per emettere richieste di I/O di dimensione 1 MB per un host Broadcom NVMe/FC, è necessario aumentare il `lpfc` valore del `lpfc_sg_seg_cnt` parametro a 256 dal valore predefinito di 64.



Questi passaggi non si applicano agli host Qlogic NVMe/FC.

Fasi

1. Impostare il `lpfc_sg_seg_cnt` parametro su 256:

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

Dovresti vedere un output simile al seguente esempio:

```
options lpfc lpfc_sg_seg_cnt=256
```

2. Eseguire il `dracut -f` comando e riavviare l'host.
3. Verificare che il valore per `lpfc_sg_seg_cnt` sia 256:

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

Passaggio 5: verificare i servizi di avvio NVMe

IL `nvmeof-boot-connections.service` E `nvme-autoconnect.service` servizi di avvio inclusi in NVMe/FC `nvme-cli` i pacchetti vengono abilitati automaticamente all'avvio del sistema.

Dopo aver completato l'avvio, verificare che `nvmeof-boot-connections.service` E `nvme-autoconnect.service` i servizi di avvio sono abilitati.

Fasi

1. Verificare che `nvme-autoconnect.service` sia attivato:

```
systemctl status nvme-autoconnect.service
```

Mostra output di esempio

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

Jun 10 04:06:26 SR630-13-201.lab.eng.btc.netapp.in systemd[1]:
Starting Connect NVMe-oF subsystems automatically during boot...
Jun 10 04:06:26 SR630-13-201.lab.eng.btc.netapp.in systemd[1]: nvme-
autoconnect.service: Deactivated successfully.
Jun 10 04:06:26 SR630-13-201.lab.eng.btc.netapp.in systemd[1]:
Finished Connect NVMe-oF subsystems automatically during boot.
```

2. Verificare che `nvmeof-boot-connections.service` sia attivato:

```
systemctl status nvme-fc-boot-connections.service
```

Mostra output di esempio

```
nvme-fc-boot-connections.service - Auto-connect to subsystems on FC-
NVME devices found during boot
   Loaded: loaded (/usr/lib/systemd/system/nvme-fc-boot-
connections.service; enabled; preset: enabled)
   Active: inactive (dead) since Tue 2025-06-10 01:08:36 EDT; 2h
59min ago
     Main PID: 7090 (code=exited, status=0/SUCCESS)
        CPU: 30ms

Jun 10 01:08:36 localhost systemd[1]: Starting Auto-connect to
subsystems on FC-NVME devices found during boot...
Jun 10 01:08:36 localhost systemd[1]: nvme-fc-boot-
connections.service: Deactivated successfully.
Jun 10 01:08:36 localhost systemd[1]: Finished Auto-connect to
subsystems on FC-NVME devices found during boot.
```

Passaggio 6: verificare la configurazione multipathing

Verificare che lo stato multipath NVMe in-kernel, lo stato ANA e i namespace ONTAP siano corretti per la configurazione NVMe-of.

Fasi

1. Verificare che il multipath NVMe nel kernel sia attivato:

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

Viene visualizzato il seguente output:

```
Y
```

2. Verificare che le impostazioni NVMe-of appropriate (ad esempio, modello impostato su controller NetApp ONTAP e ipolicy per il bilanciamento del carico impostato su round-robin) per i rispettivi spazi dei nomi ONTAP si riflettano correttamente sull'host:

- a. Visualizza i sottosistemi:

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

Viene visualizzato il seguente output:

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

b. Visualizza la politica:

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

Viene visualizzato il seguente output:

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

3. Verificare che gli spazi dei nomi siano stati creati e rilevati correttamente sull'host:

```
nvme list
```

Mostra esempio

Node	SN	Model

/dev/nvme4n1	81Ix2BVuekWcAAAAAAB	NetApp ONTAP Controller

Namespace	Usage	Format	FW	Rev

1		21.47 GB / 21.47 GB	4 KiB + 0 B	FFFFFFFF

4. Verificare che lo stato del controller di ciascun percorso sia attivo e che abbia lo stato ANA corretto:

NVMe/FC

```
nvme list-subsys /dev/nvme4n5
```

Mostra esempio

```
nvme-subsys4 - NQN=nqn.1992-08.com.netapp:sn.3a5d31f5502c11ef9f50d039eab6cb6d:subsystem.nvme_1
                hostnqn=nqn.2014-08.org.nvmexpress:uuid:e6dade64-216d-11ec-b7bb-7ed30a5482c3
iopolICY=round-robin\
+- nvme1 fc traddr=nn-0x2082d039eaa7dfc8:pn-0x2088d039eaa7dfc8,host_traddr=nn-0x20000024ff752e6d:pn-0x21000024ff752e6d live optimized
+- nvme12 fc traddr=nn-0x2082d039eaa7dfc8:pn-0x208ad039eaa7dfc8,host_traddr=nn-0x20000024ff752e6d:pn-0x21000024ff752e6d live non-optimized
+- nvme10 fc traddr=nn-0x2082d039eaa7dfc8:pn-0x2087d039eaa7dfc8,host_traddr=nn-0x20000024ff752e6c:pn-0x21000024ff752e6c live non-optimized
+- nvme3 fc traddr=nn-0x2082d039eaa7dfc8:pn-0x2083d039eaa7dfc8,host_traddr=nn-0x20000024ff752e6c:pn-0x21000024ff752e6c live optimized
```

NVMe/TCP

```
nvme list-subsys /dev/nvme1n1
```

Mostra esempio

```
nvme-subsys5 - NQN=nqn.1992-08.com.netapp:sn.0f4bale74eb611ef9f50d039eab6cb6d:subsystem.nvme_tcp_3
hostnqn=nqn.2014-08.org.nvmexpress:uuid:4c4c4544-0035-5910-804b-b5c04f444d33
iopolicy=round-robin
\
+- nvme13 tcp
traddr=192.168.2.25,trsvcid=4420,host_traddr=192.168.2.31,
src_addr=192.168.2.31 live optimized
+- nvme14 tcp
traddr=192.168.2.24,trsvcid=4420,host_traddr=192.168.2.31,
src_addr=192.168.2.31 live non-optimized
+- nvme5 tcp
traddr=192.168.1.25,trsvcid=4420,host_traddr=192.168.1.31,
src_addr=192.168.1.31 live optimized
+- nvme6 tcp
traddr=192.168.1.24,trsvcid=4420,host_traddr=192.168.1.31,
src_addr=192.168.1.31 live non-optimized
```

5. Verificare che il plug-in NetApp visualizzi i valori corretti per ciascun dispositivo dello spazio dei nomi ONTAP:

Colonna

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

Mostra esempio

Device	Vserver	Namespace Path
/dev/nvme1n1	linux_tcnvme_iscsi	/vol/tcpnvme_1_0_0/tcpnvme_ns

NSID	UUID	Size
1	5f7f630d-8ea5-407f-a490-484b95b15dd6	21.47GB

JSON

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

Mostra esempio

```
{
  "ONTAPdevices":[
    {
      "Device":"/dev/nvme1n1",
      "Vserver":"linux_tcnvme_iscsi",
      "Namespace_Path":"/vol/tcpnvme_1_0_0/tcpnvme_ns",
      "NSID":1,
      "UUID":"5f7f630d-8ea5-407f-a490-484b95b15dd6",
      "Size":"21.47GB",
      "LBA_Data_Size":4096,
      "Namespace_Size":5242880
    },
  ]
}
```

Passaggio 7: impostare l'autenticazione in-band sicura

L'autenticazione in-band sicura è supportata tramite NVMe/TCP tra un host Rocky Linux 9x e un controller ONTAP .

Ogni host o controller deve essere associato a un `DH-HMAC-CHAP` chiave per impostare l'autenticazione sicura. Una `DH-HMAC-CHAP` La chiave è una combinazione dell'`NQN` dell'host o del controller NVMe e di un segreto di autenticazione configurato dall'amministratore. Per autenticare il proprio peer, un host o un controller NVMe deve riconoscere la chiave associata al peer.

Fasi

Imposta l'autenticazione in-band sicura tramite la CLI o un file JSON di configurazione. Se è necessario specificare chiavi `dhchap` diverse per sottosistemi diversi, è necessario utilizzare un file di configurazione JSON.

CLI

Configurare l'autenticazione in banda protetta utilizzando la CLI.

1. Ottenere l'NQN dell'host:

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

2. Genera la chiave dhchap per l'host Rocky Linux 9.x.

L'output seguente descrive il `gen-dhchap-key` parametri del comando:

```
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

Nell'esempio seguente, viene generata una chiave casuale dhCHAP con HMAC impostato su 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
W0o0PJJM6yZsTeEpGkDHMHQ255+g=:
```

3. Sul controller ONTAP, aggiungere l'host e specificare entrambe le chiavi dhchap:

```
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. Un host supporta due tipi di metodi di autenticazione, unidirezionale e bidirezionale. Sull'host, connettersi al controller ONTAP e specificare le chiavi dhchap in base al metodo di autenticazione scelto:

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

5. Convalidare `nvme connect authentication` comando verificando le chiavi `dhchap` dell'host e del controller:

a. Verificare le chiavi `dhchap` dell'host:

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

Mostra output di esempio per una configurazione unidirezionale

```
cat /sys/class/nvme-subsystem/nvme-subsys1/nvme*/dhchap_secret  
DHHC-1:01:CNxTYq73T9vJk0JpOfDBZrhDCqpWBN4XVZI5WxwPgDUieHAi:  
DHHC-1:01:CNxTYq73T9vJk0JpOfDBZrhDCqpWBN4XVZI5WxwPgDUieHAi:  
DHHC-1:01:CNxTYq73T9vJk0JpOfDBZrhDCqpWBN4XVZI5WxwPgDUieHAi:  
DHHC-1:01:CNxTYq73T9vJk0JpOfDBZrhDCqpWBN4XVZI5WxwPgDUieHAi:
```

b. Verificare i tasti `dhchap` del controller:

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

Mostra output di esempio per una configurazione bidirezionale

```
cat /sys/class/nvme-subsystem/nvme-  
subsys6/nvme*/dhchap_ctrl_secret  
DHHC-  
1:03:wSpuuKbBHTzC0W9JZxMBsYd9JFV8Si9aDh22k2BR/4m852vH7KGlrJeMp  
zhmyjDWOo0PJJM6yZsTEpGkDHMHQ255+g=:  
DHHC-  
1:03:wSpuuKbBHTzC0W9JZxMBsYd9JFV8Si9aDh22k2BR/4m852vH7KGlrJeMp  
zhmyjDWOo0PJJM6yZsTEpGkDHMHQ255+g=:  
DHHC-  
1:03:wSpuuKbBHTzC0W9JZxMBsYd9JFV8Si9aDh22k2BR/4m852vH7KGlrJeMp  
zhmyjDWOo0PJJM6yZsTEpGkDHMHQ255+g=:  
DHHC-  
1:03:wSpuuKbBHTzC0W9JZxMBsYd9JFV8Si9aDh22k2BR/4m852vH7KGlrJeMp  
zhmyjDWOo0PJJM6yZsTEpGkDHMHQ255+g=:
```

JSON

Quando più sottosistemi NVMe sono disponibili sul controller ONTAP , è possibile utilizzare `/etc/nvme/config.json` file con il `nvme connect-all` comando.

Utilizzare il `-o` opzione per generare il file JSON. Per ulteriori opzioni di sintassi, fare riferimento alle pagine man di NVMe connect-all.

1. Configurare il file JSON.



Nell'esempio seguente, `dhchap_key` corrisponde a `dhchap_secret` E `dhchap_ctrl_key` corrisponde a `dhchap_ctrl_secret`.

Mostra esempio

```
cat /etc/nvme/config.json
[
{
  "hostnqn":"nqn.2014-08.org.nvmexpress:uuid:9796c1ec-0d34-11eb-
b6b2-3a68dd3bab57",
  "hostid":"b033cd4fd6db4724adb48655bfb55448",
  "dhchap_key":" DHHC-
1:01:CNxTYq73T9vJk0JpOfDBZrhDCqpWBN4XVZI5WxwPgDUieHAI:"
},
{
  "hostnqn":"nqn.2014-08.org.nvmexpress:uuid:4c4c4544-0035-5910-
804b-b5c04f444d33",
  "subsystems":[
    {
      "nqn":"nqn.1992-
08.com.netapp:sn.0f4bale74eb611ef9f50d039eab6cb6d:subsystem.bidi
r_DHCP",
      "ports":[
        {
          "transport":"tcp",
          "traddr":" 192.168.1.24 ",
          "host_traddr":" 192.168.1.31 ",
          "trsvcid":"4420",
          "dhchap_ctrl_key":"DHHC-
1:03:
wSpuuKbBHTzC0W9JZxMBsYd9JFV8Si9aDh22k2BR/4m852vH7KGlrJeMpzhmyjDW
Oo0PJJM6yZsTeEpGkDHMHQ255+g=:"
        },
        {
          "transport":"tcp",
          "traddr":" 192.168.1.25 ",
          "host_traddr":" 192.168.1.31",
          "trsvcid":"4420",
          "dhchap_ctrl_key":"DHHC-
1:03:
wSpuuKbBHTzC0W9JZxMBsYd9JFV8Si9aDh22k2BR/4m852vH7KGlrJeMpzhmyjDW
Oo0PJJM6yZsTeEpGkDHMHQ255+g=:"
        },
        {
          "transport":"tcp",
          "traddr":" 192.168.2.24 ",
          "host_traddr":" 192.168.2.31",
          "trsvcid":"4420",
```

```

        "dhchap_ctrl_key":"DHHC-
        1:03:
wSpuuKbBHTzC0W9JZxMBsYd9JFV8Si9aDh22k2BR/4m852vH7KGlrJeMpzhmyjDW
Oo0PJJM6yZsTeEpGkDHMHQ255+g="
    },
    {
        "transport":"tcp",
        "traddr":" 192.168.2.25 ",
        "host_traddr":" 192.168.2.31",
        "trsvcid":"4420",
        "dhchap_ctrl_key":"DHHC-
        1:03:
wSpuuKbBHTzC0W9JZxMBsYd9JFV8Si9aDh22k2BR/4m852vH7KGlrJeMpzhmyjDW
Oo0PJJM6yZsTeEpGkDHMHQ255+g="
    }
]
}
]

```

2. Connettersi al controller ONTAP utilizzando il file di configurazione JSON:

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

Mostra esempio

```
already connected to hostnqn=nqn.2014-
08.org.nvmexpress:uuid:4c4c4544-0035-5910-804b-
b5c04f444d33,nqn=nqn.1992-
08.com.netapp:sn.8dde3be2cc7c11efb777d039eab6cb6d:subsystem.
bidi
r_DHCP,transport=tcp,traddr=192.168.1.25,trsvcid=4420
already connected to hostnqn=nqn.2014-
08.org.nvmexpress:uuid:4c4c4544-0035-5910-804b-
b5c04f444d33,nqn=nqn.1992-
08.com.netapp:sn.8dde3be2cc7c11efb777d039eab6cb6d:subsystem.
bidi
r_DHCP,transport=tcp,traddr=192.168.2.25,trsvcid=4420
already connected to hostnqn=nqn.2014-
08.org.nvmexpress:uuid:4c4c4544-0035-5910-804b-
b5c04f444d33,nqn=nqn.1992-
08.com.netapp:sn.8dde3be2cc7c11efb777d039eab6cb6d:subsystem.
bidi
r_DHCP,transport=tcp,traddr=192.168.1.24,trsvcid=4420
already connected to hostnqn=nqn.2014-
08.org.nvmexpress:uuid:4c4c4544-0035-5910-804b-
b5c04f444d33,nqn=nqn.1992-
08.com.netapp:sn.8dde3be2cc7c11efb777d039eab6cb6d:subsystem.
bidi
r_DHCP,transport=tcp,traddr=192.168.2.24,trsvcid=4420
```

3. Verificare che i segreti dhchap siano stati abilitati per i rispettivi controller per ciascun sottosistema:

a. Verificare le chiavi dhchap dell'host:

```
cat /sys/class/nvme-subsystem/nvme-subsys0/nvme0/dhchap_secret
```

L'esempio seguente mostra una chiave dhchap:

```
DHHC-1:01:CNxTYq73T9vJk0JpOfDBZrhDCqpWBN4XVZI5WxwPgDUieHAI:
```

b. Verificare i tasti dhchap del controller:

```
cat /sys/class/nvme-subsystem/nvme-
subsys0/nvme0/dhchap_ctrl_secret
```

Dovresti vedere un output simile al seguente esempio:

```
DHHC-  
1:03:wSpuuKbBHTzC0W9JZxMBsYd9JFV8Si9aDh22k2BR/4m852vH7KGlrJeMpzhmyjD  
WOo0PJJM6yZsTeEpGkDHMHQ255+g=:
```

Fase 8: Esaminare i problemi noti

Non ci sono problemi noti.

Configurare Rocky Linux 8.x per NVMe-oF con storage ONTAP

Gli host Rocky Linux supportano i protocolli NVMe su Fibre Channel (NVMe/FC) e NVMe su TCP (NVMe/TCP) con Asymmetric Namespace Access (ANA). ANA fornisce funzionalità multipathing equivalenti all'accesso asimmetrico alle unità logiche (ALUA) negli ambienti iSCSI e FCP.

Scopri come configurare gli host NVMe over Fabrics (NVMe-oF) per Rocky Linux 8.x. Per ulteriori informazioni sul supporto e sulle funzionalità, vedere ["Supporto e funzionalità di Rocky Linux ONTAP"](#).

NVMe-oF con Rocky Linux 8.x presenta le seguenti limitazioni note:

- L'avvio SAN tramite il protocollo NVMe-oF non è attualmente supportato.
- Il multipath NVMe nel kernel è disabilitato per impostazione predefinita sugli host NVMe-oF in Rocky Linux 8.x; è necessario abilitarlo manualmente.
- NVMe/TCP è disponibile come anteprima tecnologica a causa di problemi noti.

Passaggio 1: installa Rocky Linux e il software NVMe e verifica la configurazione

Per configurare l'host per NVMe-oF è necessario installare i pacchetti software host e NVMe, abilitare il multipathing e verificare la configurazione NQN dell'host.

Fasi

1. Installare Rocky Linux 8.x sul server. Una volta completata l'installazione, verifica di avere installato il kernel Rocky Linux 8.x richiesto:

```
uname -r
```

Esempio di versione del kernel Rocky Linux:

```
5.14.0-570.12.1.el9_6.x86_64
```

2. Installare `nvme-cli` pacchetto:

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

L'esempio seguente mostra una versione del pacchetto nvme-cli:

```
nvme-cli-2.11-5.el9.x86_64
```

3. Installare libnvme pacchetto:

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

L'esempio seguente mostra un libnvme versione del pacchetto:

```
libnvme-1.11.1-1.el9.x86_64
```

4. Sull'host Rocky Linux, controlla la stringa hostnqn in /etc/nvme/hostnqn :

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

L'esempio seguente mostra un hostnqn versione:

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

5. Sul sistema ONTAP , verificare che hostnqn la stringa corrisponde a hostnqn stringa per il sottosistema corrispondente nell'array ONTAP :

```
::> vserver nvme subsystem host show -vserver vs_coexistence_LPE36002
```

Mostra esempio

```
Vserver Subsystem Priority Host NQN
-----
vs_coexistence_LPE36002
    nvme
        regular    nqn.2014-
08.org.nvmexpress:uuid:4c4c4544-0056-5410-8048-b9c04f425633
    nvme_1
        regular    nqn.2014-
08.org.nvmexpress:uuid:4c4c4544-0056-5410-8048-b9c04f425633
    nvme_2
        regular    nqn.2014-
08.org.nvmexpress:uuid:4c4c4544-0056-5410-8048-b9c04f425633
    nvme_3
        regular    nqn.2014-
08.org.nvmexpress:uuid:4c4c4544-0056-5410-8048-b9c04f425633
4 entries were displayed.
```



Se il `hostnqn` le stringhe non corrispondono, utilizzare `vserver modify` per aggiornare `hostnqn` Stringa sul sottosistema di array ONTAP corrispondente a `hostnqn` stringa da `/etc/nvme/hostnqn` sull'host.

Passaggio 2: configurare NVMe/FC e NVMe/TCP

Configurare NVMe/FC con adattatori Broadcom/Emulex o Marvell/QLogic oppure configurare NVMe/TCP utilizzando operazioni di rilevamento e connessione manuali.

NVMe/FC - Broadcom/Emulex

Configurare NVMe/FC per un adattatore Broadcom/Emulex.

Fasi

1. Verificare di utilizzare il modello di adattatore supportato:

a. Visualizza i nomi dei modelli:

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

Viene visualizzato il seguente output:

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

b. Visualizza le descrizioni dei modelli:

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

Dovresti vedere un output simile al seguente esempio:

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

2. Verificare di utilizzare il Broadcom consigliato lpfc firmware e driver della posta in arrivo:

a. Visualizza la versione del firmware:

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

Il comando restituisce le versioni del firmware:

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

b. Visualizza la versione del driver in arrivo:

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

L'esempio seguente mostra la versione del driver:

```
0:14.4.0.2
```

Per l'elenco aggiornato dei driver della scheda di rete supportati e delle versioni del firmware, vedere ["Tool di matrice di interoperabilità"](#).

3. Verificare che l'output previsto di `lpfc_enable_fc4_type` sia impostato su 3:

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

4. Verificare che sia possibile visualizzare le porte dell'iniziatore:

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

L'esempio seguente mostra le identità delle porte:

```
0x100000109bf044b1  
0x100000109bf044b2
```

5. Verificare che le porte dell'iniziatore siano in linea:

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

Viene visualizzato il seguente output:

```
Online  
Online
```

6. Verificare che le porte iniziatore NVMe/FC siano abilitate e che le porte di destinazione siano visibili:

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

Mostra esempio

```
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 x202fd039eaa7dfc8 WWNN x202cd039eaa7dfc8
DID x021310 TARGET DISCSRV ONLINE
NVME RPORT          WWPN x202dd039eaa7dfc8 WWNN x202cd039eaa7dfc8
DID x020b10 TARGET DISCSRV ONLINE
```

```
NVME Statistics
LS: Xmt 0000000810 Cmpl 0000000810 Abort 00000000
LS XMIT: Err 00000000 CMPL: xb 00000000 Err 00000000
Total FCP Cmpl 000000007b098f07 Issue 000000007aee27c4 OutIO
ffffffffffffe498bd
          abort 000013b4 noxri 00000000 nondlp 00000058 qdepth
00000000 wqerr 00000000 err 00000000
FCP CMPL: xb 000013b4 Err 00021443
```

```
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 x2033d039eaa7dfc8 WWNN x202cd039eaa7dfc8
DID x020110 TARGET DISCSRV ONLINE
NVME RPORT          WWPN x2032d039eaa7dfc8 WWNN x202cd039eaa7dfc8
DID x022910 TARGET DISCSRV ONLINE
```

```
NVME Statistics
LS: Xmt 0000000840 Cmpl 0000000840 Abort 00000000
LS XMIT: Err 00000000 CMPL: xb 00000000 Err 00000000
Total FCP Cmpl 000000007afd4434 Issue 000000007ae31b83 OutIO
ffffffffffffe5d74f
          abort 000014a5 noxri 00000000 nondlp 0000006a qdepth
00000000 wqerr 00000000 err 00000000
FCP CMPL: xb 000014a5 Err 0002149a
```

NVMe/FC - Marvell/QLogic

Configurare NVMe/FC per un adattatore Marvell/QLogic.

Fasi

1. Verificare che siano in esecuzione le versioni del firmware e del driver dell'adattatore supportate:

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

L'esempio seguente mostra le versioni del driver e del firmware:

```
QLE2742 FW:v9.14.00 DVR:v10.02.09.200-k  
QLE2742 FW:v9.14.00 DVR:v10.02.09.200-k
```

2. Verificare che `ql2xnvmeenable` è impostato. Ciò consente all'adattatore Marvell di funzionare come iniziatore NVMe/FC:

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

L'uscita prevista è 1.

NVMe/TCP

Il protocollo NVMe/TCP non supporta l'operazione di connessione automatica. In alternativa, è possibile scoprire i sottosistemi e gli spazi dei nomi NVMe/TCP eseguendo l'`NVMe/TCP connect O connect-all` operazioni manualmente.

Fasi

1. Verificare che la porta di avvio possa ottenere i dati della pagina del registro di individuazione attraverso i LIF NVMe/TCP supportati:

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

Mostra esempio

```
nvme discover -t tcp -w 192.168.1.31 -a 192.168.1.24
Discovery Log Number of Records 20, Generation counter 25
=====Discovery Log Entry 0=====
trtype: tcp
adrfam: ipv4
subtype: current discovery subsystem
treq: not specified
portid: 4
trsvcid: 8009
subnqn: nqn.1992-
08.com.netapp:sn.0f4bale74eb611ef9f50d039eab6cb6d:discovery
traddr: 192.168.2.25
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: 2
trsvcid: 8009
subnqn: nqn.1992-
08.com.netapp:sn.0f4bale74eb611ef9f50d039eab6cb6d:discovery
traddr: 192.168.1.25
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: 5
trsvcid: 8009
subnqn: nqn.1992-
08.com.netapp:sn.0f4bale74eb611ef9f50d039eab6cb6d:discovery
traddr: 192.168.2.24
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:  1
trsvcid: 8009
subnqn:  nqn.1992-
08.com.netapp:sn.0f4bale74eb611ef9f50d039eab6cb6d:discovery
traddr:  192.168.1.24
eflags:  explicit discovery connections, duplicate discovery
information
sectype: none
=====Discovery Log Entry 4=====
trtype:  tcp
adrfam:  ipv4
subtype: nvme subsystem
treq:    not specified
portid:  4
trsvcid: 4420
subnqn:  nqn.1992-
08.com.netapp:sn.0f4bale74eb611ef9f50d039eab6cb6d:subsystem.nvme
_tcp_1
traddr:  192.168.2.25
eflags:  none
sectype: none
=====Discovery Log Entry 5=====
trtype:  tcp
adrfam:  ipv4
subtype: nvme subsystem
treq:    not specified
portid:  2
trsvcid: 4420
subnqn:  nqn.1992-
08.com.netapp:sn.0f4bale74eb611ef9f50d039eab6cb6d:subsystem.nvme
_tcp_1
traddr:  192.168.1.25
eflags:  none
sectype: none
=====Discovery Log Entry 6=====
trtype:  tcp
adrfam:  ipv4
subtype: nvme subsystem
treq:    not specified
portid:  5
trsvcid: 4420
subnqn:  nqn.1992-
08.com.netapp:sn.0f4bale74eb611ef9f50d039eab6cb6d:subsystem.nvme
_tcp_1

```

```
traddr: 192.168.2.24
eflags: none
sectype: none
=====Discovery Log Entry 7=====
trtype: tcp
adrfam: ipv4
subtype: nvme subsystem
treq: not specified
portid: 1
trsvcid: 4420
subnqn: nqn.1992-
08.com.netapp:sn.0f4bale74eb611ef9f50d039eab6cb6d:subsystem.nvme
_tcp_1
traddr: 192.168.1.24
eflags: none
sectype: none
=====Discovery Log Entry 8=====
trtype: tcp
adrfam: ipv4
subtype: nvme subsystem
treq: not specified
portid: 4
trsvcid: 4420
subnqn: nqn.1992-
08.com.netapp:sn.0f4bale74eb611ef9f50d039eab6cb6d:subsystem.nvme
_tcp_4
traddr: 192.168.2.25
eflags: none
sectype: none
=====Discovery Log Entry 9=====
trtype: tcp
adrfam: ipv4
subtype: nvme subsystem
treq: not specified
portid: 2
trsvcid: 4420
subnqn: nqn.1992-
08.com.netapp:sn.0f4bale74eb611ef9f50d039eab6cb6d:subsystem.nvme
_tcp_4
traddr: 192.168.1.25
eflags: none
sectype: none
=====Discovery Log Entry 10=====
trtype: tcp
adrfam: ipv4
subtype: nvme subsystem
```

```

treq:    not specified
portid:  5
trsvcid: 4420
subnqn:  nqn.1992-
08.com.netapp:sn.0f4ba1e74eb611ef9f50d039eab6cb6d:subsystem.nvme
_tcp_4
traddr:  192.168.2.24
eflags:  none
sectype: none
=====Discovery Log Entry 11=====
trtype:  tcp
adrfam:  ipv4
subtype: nvme subsystem
treq:    not specified
portid:  1
trsvcid: 4420
subnqn:  nqn.1992-
08.com.netapp:sn.0f4ba1e74eb611ef9f50d039eab6cb6d:subsystem.nvme
_tcp_4
traddr:  192.168.1.24
eflags:  none
sectype: none
=====Discovery Log Entry 12=====
trtype:  tcp
adrfam:  ipv4
subtype: nvme subsystem
treq:    not specified
portid:  4
trsvcid: 4420
subnqn:  nqn.1992-
08.com.netapp:sn.0f4ba1e74eb611ef9f50d039eab6cb6d:subsystem.nvme
_tcp_3
traddr:  192.168.2.25
eflags:  none
sectype: none
=====Discovery Log Entry 13=====
trtype:  tcp
adrfam:  ipv4
subtype: nvme subsystem
treq:    not specified
portid:  2
trsvcid: 4420
subnqn:  nqn.1992-
08.com.netapp:sn.0f4ba1e74eb611ef9f50d039eab6cb6d:subsystem.nvme
_tcp_3
traddr:  192.168.1.25

```

```

eflags: none
sectype: none
=====Discovery Log Entry 14=====
trtype: tcp
adrfam: ipv4
subtype: nvme subsystem
treq: not specified
portid: 5
trsvcid: 4420
subnqn: nqn.1992-
08.com.netapp:sn.0f4ba1e74eb611ef9f50d039eab6cb6d:subsystem.nvme
_tcp_3
traddr: 192.168.2.24
eflags: none
sectype: none
=====Discovery Log Entry 15=====
trtype: tcp
adrfam: ipv4
subtype: nvme subsystem
treq: not specified
portid: 1
trsvcid: 4420
subnqn: nqn.1992-
08.com.netapp:sn.0f4ba1e74eb611ef9f50d039eab6cb6d:subsystem.nvme
_tcp_3
traddr: 192.168.1.24
eflags: none
sectype: none
=====Discovery Log Entry 16=====
trtype: tcp
adrfam: ipv4
subtype: nvme subsystem
treq: not specified
portid: 4
trsvcid: 4420
subnqn: nqn.1992-
08.com.netapp:sn.0f4ba1e74eb611ef9f50d039eab6cb6d:subsystem.nvme
_tcp_2
traddr: 192.168.2.25
eflags: none
sectype: none
=====Discovery Log Entry 17=====
trtype: tcp
adrfam: ipv4
subtype: nvme subsystem
treq: not specified

```

```

portid: 2
trsvcid: 4420
subnqn: nqn.1992-
08.com.netapp:sn.0f4bale74eb611ef9f50d039eab6cb6d:subsystem.nvme
_tcp_2
traddr: 192.168.1.25
eflags: none
sectype: none
=====Discovery Log Entry 18=====
trtype: tcp
adrfam: ipv4
subtype: nvme subsystem
treq: not specified
portid: 5
trsvcid: 4420
subnqn: nqn.1992-
08.com.netapp:sn.0f4bale74eb611ef9f50d039eab6cb6d:subsystem.nvme
_tcp_2
traddr: 192.168.2.24
eflags: none
sectype: none
=====Discovery Log Entry 19=====
trtype: tcp
adrfam: ipv4
subtype: nvme subsystem
treq: not specified
portid: 1
trsvcid: 4420
subnqn: nqn.1992-
08.com.netapp:sn.0f4bale74eb611ef9f50d039eab6cb6d:subsystem.nvme
_tcp_2
traddr: 192.168.1.24
eflags: none
sectype: none

```

2. Verifica che le altre combinazioni di LIF initiator NVMe/TCP siano in grado di recuperare correttamente i dati della pagina del log di rilevamento:

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

Mostra esempio

```
nvme discover -t tcp -w 192.168.1.31 -a 192.168.1.24
nvme discover -t tcp -w 192.168.2.31 -a 192.168.2.24
nvme discover -t tcp -w 192.168.1.31 -a 192.168.1.25
nvme discover -t tcp -w 192.168.2.31 -a 192.168.2.25
```

3. Eseguire `nvme connect-all` Command tra tutti i LIF target initiator NVMe/TCP supportati nei nodi:

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

Mostra esempio

```
nvme      connect-all -t tcp -w 192.168.1.31 -a 192.168.1.24
nvme      connect-all -t tcp -w 192.168.2.31 -a 192.168.2.24
nvme      connect-all -t tcp -w 192.168.1.31 -a 192.168.1.25
nvme      connect-all -t tcp -w 192.168.2.31 -a 192.168.2.25
```

Passaggio 3: Facoltativamente, abilitare 1 MB di I/O per NVMe/FC

È possibile abilitare richieste di I/O di dimensione pari a 1 MB per NVMe/FC configurate con una scheda Broadcom. ONTAP segnala una dimensione massima di trasferimento dati (MDTS) di 8 nei dati di Identify Controller. Ciò significa che le dimensioni massime delle richieste i/o possono essere fino a 1MB MB. Per inviare richieste di I/O di dimensione pari a 1 MB, è necessario aumentare il valore `lpfc` del parametro `lpfc_sg_seg_cnt` parametro a 256 dal valore predefinito di 64.



Questi passaggi non si applicano agli host Qlogic NVMe/FC.

Fasi

1. Impostare il `lpfc_sg_seg_cnt` parametro su 256:

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

```
options lpfc lpfc_sg_seg_cnt=256
```

2. Eseguire il `dracut -f` comando e riavviare l'host.
3. Verificare che il valore per `lpfc_sg_seg_cnt` sia 256:

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

Passaggio 4: verificare la configurazione del multipathing

Verificare che lo stato multipath NVMe in-kernel, lo stato ANA e i namespace ONTAP siano corretti per la configurazione NVMe-of.

Fasi

1. Verificare che il multipath NVMe nel kernel sia attivato:

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

Viene visualizzato il seguente output:

```
Y
```

2. Verificare che le impostazioni NVMe-of appropriate (ad esempio, modello impostato su controller NetApp ONTAP e ipolicy per il bilanciamento del carico impostato su round-robin) per i rispettivi spazi dei nomi ONTAP si riflettano correttamente sull'host:

- a. Visualizza i sottosistemi:

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

Viene visualizzato il seguente output:

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

- b. Visualizza la politica:

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

Viene visualizzato il seguente output:

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

3. Verificare che gli spazi dei nomi siano stati creati e rilevati correttamente sull'host:

```
nvme list
```

Mostra esempio

```
Node          SN                      Model
-----
/dev/nvme4n1  81Ix2BVuekWcAAAAAAB  NetApp ONTAP Controller

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

4. Verificare che lo stato del controller di ciascun percorso sia attivo e che abbia lo stato ANA corretto:

NVMe/FC

```
nvme list-subsys /dev/nvme4n5
```

Mostra esempio

```
nvme-subsys4 - NQN=nqn.1992-08.com.netapp:sn.3a5d31f5502c11ef9f50d039eab6cb6d:subsystem.nvme_1
                hostnqn=nqn.2014-08.org.nvmexpress:uuid:e6dade64-216d-11ec-b7bb-7ed30a5482c3
iopolicy=round-robin\
+- nvme1 fc traddr=nn-0x2082d039eaa7dfc8:pn-0x2088d039eaa7dfc8,host_traddr=nn-0x20000024ff752e6d:pn-0x21000024ff752e6d live optimized
+- nvme12 fc traddr=nn-0x2082d039eaa7dfc8:pn-0x208ad039eaa7dfc8,host_traddr=nn-0x20000024ff752e6d:pn-0x21000024ff752e6d live non-optimized
+- nvme10 fc traddr=nn-0x2082d039eaa7dfc8:pn-0x2087d039eaa7dfc8,host_traddr=nn-0x20000024ff752e6c:pn-0x21000024ff752e6c live non-optimized
+- nvme3 fc traddr=nn-0x2082d039eaa7dfc8:pn-0x2083d039eaa7dfc8,host_traddr=nn-0x20000024ff752e6c:pn-0x21000024ff752e6c live optimized
```

NVMe/TCP

```
nvme list-subsys /dev/nvme1n1
```

Mostra esempio

```
nvme-subsys5 - NQN=nqn.1992-  
08.com.netapp:sn.0f4bale74eb611ef9f50d039eab6cb6d:subsystem.nvme  
_tcp_3  
hostnqn=nqn.2014-08.org.nvmexpress:uuid:4c4c4544-0035-5910-804b-  
b5c04f444d33  
iopolicy=round-robin  
\  
+- nvme13 tcp  
traddr=192.168.2.25,trsvcid=4420,host_traddr=192.168.2.31,  
src_addr=192.168.2.31 live optimized  
+- nvme14 tcp  
traddr=192.168.2.24,trsvcid=4420,host_traddr=192.168.2.31,  
src_addr=192.168.2.31 live non-optimized  
+- nvme5 tcp  
traddr=192.168.1.25,trsvcid=4420,host_traddr=192.168.1.31,  
src_addr=192.168.1.31 live optimized  
+- nvme6 tcp  
traddr=192.168.1.24,trsvcid=4420,host_traddr=192.168.1.31,  
src_addr=192.168.1.31 live non-optimized
```

5. Verificare che il plug-in NetApp visualizzi i valori corretti per ciascun dispositivo dello spazio dei nomi ONTAP:

Colonna

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

Mostra esempio

Device	Vserver	Namespace Path
/dev/nvme1n1	linux_tcnvme_iscsi	/vol/tcpcnvme_1_0_0/tcpcnvme_ns

NSID	UUID	Size
1	5f7f630d-8ea5-407f-a490-484b95b15dd6	21.47GB

JSON

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

Mostra esempio

```
{
  "ONTAPdevices":[
    {
      "Device":"/dev/nvme1n1",
      "Vserver":"linux_tcnvme_iscsi",
      "Namespace_Path":"/vol/tcpcnvme_1_0_0/tcpcnvme_ns",
      "NSID":1,
      "UUID":"5f7f630d-8ea5-407f-a490-484b95b15dd6",
      "Size":"21.47GB",
      "LBA_Data_Size":4096,
      "Namespace_Size":5242880
    },
  ]
}
```

Passaggio 5: rivedere i problemi noti

Ecco i problemi noti:

ID bug NetApp	Titolo	Descrizione
"1479047"	Gli host Rocky Linux 8.x NVMe-oF creano controller di rilevamento persistenti duplicati	Sugli host NVMe-oF, è possibile utilizzare il comando "nvme discover -p" per creare Persistent Discovery Controller (PDC). Tuttavia, se si esegue Rocky Linux 8.x su un host NVMe-oF, viene creato un PDC duplicato ogni volta che viene eseguito "nvme discover -p". Quando si utilizza questo comando, è necessario creare un solo PDC per ogni combinazione iniziatore-destinazione. Tuttavia, se si esegue Rocky Linux 8.x su un host NVMe-oF, viene creato un PDC duplicato ogni volta che viene eseguito "nvme discover -p". Ciò comporta un utilizzo non necessario delle risorse sia sull'host che sulla destinazione.

Informazioni sul copyright

Copyright © 2026 NetApp, Inc. Tutti i diritti riservati. Stampato negli Stati Uniti d'America. Nessuna porzione di questo documento soggetta a copyright può essere riprodotta in qualsiasi formato o mezzo (grafico, elettronico o meccanico, inclusi fotocopie, registrazione, nastri o storage in un sistema elettronico) senza previo consenso scritto da parte del detentore del copyright.

Il software derivato dal materiale sottoposto a copyright di NetApp è soggetto alla seguente licenza e dichiarazione di non responsabilità:

IL PRESENTE SOFTWARE VIENE FORNITO DA NETAPP "COSÌ COM'È" E SENZA QUALSIVOGLIA TIPO DI GARANZIA IMPLICITA O ESPRESSA FRA CUI, A TITOLO ESEMPLIFICATIVO E NON ESAUSTIVO, GARANZIE IMPLICITE DI COMMERCIALIZZABILITÀ E IDONEITÀ PER UNO SCOPO SPECIFICO, CHE VENGONO DECLINATE DAL PRESENTE DOCUMENTO. NETAPP NON VERRÀ CONSIDERATA RESPONSABILE IN ALCUN CASO PER QUALSIVOGLIA DANNO DIRETTO, INDIRETTO, ACCIDENTALE, SPECIALE, ESEMPLARE E CONSEGUENZIALE (COMPRESI, A TITOLO ESEMPLIFICATIVO E NON ESAUSTIVO, PROCUREMENT O SOSTITUZIONE DI MERCI O SERVIZI, IMPOSSIBILITÀ DI UTILIZZO O PERDITA DI DATI O PROFITTI OPPURE INTERRUZIONE DELL'ATTIVITÀ AZIENDALE) CAUSATO IN QUALSIVOGLIA MODO O IN RELAZIONE A QUALUNQUE TEORIA DI RESPONSABILITÀ, SIA ESSA CONTRATTUALE, RIGOROSA O DOVUTA A INSOLVENZA (COMPRESA LA NEGLIGENZA O ALTRO) INSORTA IN QUALSIASI MODO ATTRAVERSO L'UTILIZZO DEL PRESENTE SOFTWARE ANCHE IN PRESENZA DI UN PREAVVISO CIRCA L'EVENTUALITÀ DI QUESTO TIPO DI DANNI.

NetApp si riserva il diritto di modificare in qualsiasi momento qualunque prodotto descritto nel presente documento senza fornire alcun preavviso. NetApp non si assume alcuna responsabilità circa l'utilizzo dei prodotti o materiali descritti nel presente documento, con l'eccezione di quanto concordato espressamente e per iscritto da NetApp. L'utilizzo o l'acquisto del presente prodotto non comporta il rilascio di una licenza nell'ambito di un qualche diritto di brevetto, marchio commerciale o altro diritto di proprietà intellettuale di NetApp.

Il prodotto descritto in questa guida può essere protetto da uno o più brevetti degli Stati Uniti, esteri o in attesa di approvazione.

LEGENDA PER I DIRITTI SOTTOPOSTI A LIMITAZIONE: l'utilizzo, la duplicazione o la divulgazione da parte degli enti governativi sono soggetti alle limitazioni indicate nel sottoparagrafo (b)(3) della clausola Rights in Technical Data and Computer Software del DFARS 252.227-7013 (FEB 2014) e FAR 52.227-19 (DIC 2007).

I dati contenuti nel presente documento riguardano un articolo commerciale (secondo la definizione data in FAR 2.101) e sono di proprietà di NetApp, Inc. Tutti i dati tecnici e il software NetApp forniti secondo i termini del presente Contratto sono articoli aventi natura commerciale, sviluppati con finanziamenti esclusivamente privati. Il governo statunitense ha una licenza irrevocabile limitata, non esclusiva, non trasferibile, non cedibile, mondiale, per l'utilizzo dei Dati esclusivamente in connessione con e a supporto di un contratto governativo statunitense in base al quale i Dati sono distribuiti. Con la sola esclusione di quanto indicato nel presente documento, i Dati non possono essere utilizzati, divulgati, riprodotti, modificati, visualizzati o mostrati senza la previa approvazione scritta di NetApp, Inc. I diritti di licenza del governo degli Stati Uniti per il Dipartimento della Difesa sono limitati ai diritti identificati nella clausola DFARS 252.227-7015(b) (FEB 2014).

Informazioni sul marchio commerciale

NETAPP, il logo NETAPP e i marchi elencati alla pagina <http://www.netapp.com/TM> sono marchi di NetApp, Inc. Gli altri nomi di aziende e prodotti potrebbero essere marchi dei rispettivi proprietari.