



Linux rocoso

ONTAP SAN Host Utilities

NetApp
January 30, 2026

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

Tabla de contenidos

- Linux rocoso 1
 - Obtenga información sobre la compatibilidad y las características de ONTAP para Rocky Linux 1
 - ¿Qué sigue? 1
 - Configurar Rocky Linux 10.x para NVMe-oF con almacenamiento ONTAP 2
 - Paso 1: Opcionalmente, habilite el arranque SAN 2
 - Paso 2: Instale el software Rocky Linux y NVMe y verifique su configuración 2
 - Paso 3: Configurar NVMe/FC y NVMe/TCP 4
 - Paso 4: Opcionalmente, habilite 1 MB de E/S para NVMe/FC 12
 - Paso 5: Verificar los servicios de arranque NVMe 13
 - Paso 6: Verificar la configuración de rutas múltiples 14
 - Paso 7: Configurar la autenticación segura en banda 18
 - Paso 8: Revise los problemas conocidos 25
 - Configurar Rocky Linux 9.x para NVMe-oF con almacenamiento ONTAP 25
 - Paso 1: Opcionalmente, habilite el arranque SAN 25
 - Paso 2: Instale el software Rocky Linux y NVMe y verifique su configuración 25
 - Paso 3: Configurar NVMe/FC y NVMe/TCP 27
 - Paso 4: Opcionalmente, habilite 1 MB de E/S para NVMe/FC 39
 - Paso 5: Verificar los servicios de arranque NVMe 40
 - Paso 6: Verificar la configuración de rutas múltiples 41
 - Paso 7: Configurar la autenticación segura en banda 45
 - Paso 8: Revise los problemas conocidos 53
 - Configurar Rocky Linux 8.x para NVMe-oF con almacenamiento ONTAP 53
 - Paso 1: Instale el software Rocky Linux y NVMe y verifique su configuración 53
 - Paso 2: Configurar NVMe/FC y NVMe/TCP 55
 - Paso 3: Opcionalmente, habilite 1 MB de E/S para NVMe/FC 66
 - Paso 4: Verificar la configuración de rutas múltiples 67
 - Paso 5: Revisar los problemas conocidos 71

Linux rocoso

Obtenga información sobre la compatibilidad y las características de ONTAP para Rocky Linux

Las características compatibles con la configuración del host con NVMe over Fabrics (NVMe-oF) varían según su versión de ONTAP y Rocky Linux.

Característica	Versión del host Rocky Linux	Versión ONTAP
Se admite la autenticación segura en banda a través de NVMe/TCP entre un host RHEL y un controlador ONTAP	9.3 o posterior	9.12.1 o posterior
NVMe/TCP proporciona espacios de nombres utilizando el protocolo nativo <code>nvme-cli</code> paquete	8.2 o posterior	9.10.1 o posterior
NVMe/TCP es una función empresarial totalmente compatible	9.0 o posterior	9.10.1 o posterior
Se admite el tráfico NVMe y SCSI en el mismo host mediante NVMe multipath para espacios de nombres NVMe-oF y dm-multipath para LUN SCSI	8.2 o posterior	9.4 o posterior

ONTAP admite las siguientes funciones de host SAN independientemente de la versión de ONTAP que se ejecute en la configuración de su sistema.

Característica	Versión del host Rocky Linux
La multirruta nativa de NVMe está habilitada de forma predeterminada	10.0 o posterior
El arranque SAN se habilita mediante el protocolo NVMe/FC	9.4 o posterior
El <code>nvme-cli</code> El paquete incluye scripts de conexión automática que eliminan la necesidad de scripts de terceros.	8.2 o posterior
La regla nativa udev en el paquete <code>nvme-cli</code> proporciona equilibrio de carga round-robin para multipathing NVMe	8.2 o posterior



Para obtener detalles sobre las configuraciones admitidas, consulte la "[Herramienta de matriz de interoperabilidad](#)".

¿Qué sigue?

Si tu versión de Rocky Linux es ..	Conozca más sobre..
Serie 10	"Configuración de NVMe para Rocky Linux 10.x"
Serie 9	"Configuración de NVMe para Rocky Linux 9.x"

Si tu versión de Rocky Linux es ..	Conozca más sobre..
Serie 8	"Configuración de NVMe para Rocky Linux 8.x"

Información relacionada

["Obtenga información sobre la gestión de protocolos NVMe"](#)

Configurar Rocky Linux 10.x para NVMe-oF con almacenamiento ONTAP

Los hosts Rocky Linux admiten los protocolos NVMe sobre canal de fibra (NVMe/FC) y NVMe sobre TCP (NVMe/TCP) con acceso asimétrico al espacio de nombres (ANA). ANA proporciona una funcionalidad de múltiples rutas equivalente al acceso a unidad lógica asimétrica (ALUA) en entornos iSCSI y FCP.

Aprenda a configurar hosts NVMe over Fabrics (NVMe-oF) para Rocky Linux 10.x. Para obtener más información sobre asistencia y funciones, consulte ["Compatibilidad y características de Rocky Linux ONTAP"](#).

NVMe-oF con Rocky Linux 10.x tiene las siguientes limitaciones conocidas:

- El `nvme disconnect-all` El comando desconecta los sistemas de archivos raíz y de datos y podría provocar inestabilidad del sistema. No emita esto en sistemas que arrancan desde SAN a través de espacios de nombres NVMe-TCP o NVMe-FC.

Paso 1: Opcionalmente, habilite el arranque SAN

Puede configurar su host para utilizar el arranque SAN para simplificar la implementación y mejorar la escalabilidad. Utilice el ["Herramienta de matriz de interoperabilidad"](#) para verificar que su sistema operativo Linux, el adaptador de bus de host (HBA), el firmware del HBA, el BIOS de arranque del HBA y la versión de ONTAP admitan el arranque SAN.

Pasos

1. ["Cree un espacio de nombres NVMe y asígnelo al host"](#) .
2. Habilite el arranque SAN en el BIOS del servidor para los puertos a los que está asignado el espacio de nombres de arranque SAN.

Para obtener información acerca de cómo activar el BIOS HBA, consulte la documentación específica de su proveedor.

3. Reinicie el host y verifique que el sistema operativo esté funcionando.

Paso 2: Instale el software Rocky Linux y NVMe y verifique su configuración

Para configurar su host para NVMe-oF, debe instalar los paquetes de software de host y NVMe, habilitar rutas múltiples y verificar la configuración NQN de su host.

Pasos

1. Instalar Rocky Linux 10.x en el servidor. Una vez completada la instalación, verifique que esté ejecutando el kernel Rocky Linux 10.x requerido:

```
uname -r
```

Ejemplo de versión del kernel de Rocky Linux:

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

2. Instale el `nvme-cli` paquete:

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

El siguiente ejemplo muestra un `nvme-cli` versión del paquete:

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

3. Instale el `libnvme` paquete:

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

El siguiente ejemplo muestra un `libnvme` versión del paquete:

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

4. En el host, verifique la cadena `hostnqn` en `/etc/nvme/hostnqn` :

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

El siguiente ejemplo muestra una `hostnqn` valor:

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

5. En el sistema ONTAP , verifique que `hostnqn` La cadena coincide con la `hostnqn` cadena para el subsistema correspondiente en la matriz ONTAP :

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

Muestra el ejemplo

```
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.
```



Si el `hostnqn` Las cadenas no coinciden, utilice el `vserver modify` comando para actualizar el `hostnqn` cadena en el subsistema del sistema de almacenamiento ONTAP correspondiente para que coincida con el `hostnqn` cadena de `/etc/nvme/hostnqn` en el host.

Paso 3: Configurar NVMe/FC y NVMe/TCP

Configure NVMe/FC con adaptadores Broadcom/Emulex o Marvell/QLogic, o configure NVMe/TCP mediante operaciones de descubrimiento y conexión manuales.

NVMe/FC - Broadcom/Emulex

Configuración de NVMe/FC para un adaptador Broadcom/Emulex.

Pasos

1. Compruebe que está utilizando el modelo de adaptador admitido:

a. Mostrar los nombres de los modelos:

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

Debe ver la siguiente salida:

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

b. Mostrar las descripciones del modelo:

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

Debería ver un resultado similar al siguiente ejemplo:

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

2. Compruebe que está utilizando la Broadcom recomendada lpfc firmware y controlador de bandeja de entrada:

a. Mostrar la versión del firmware:

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

El comando devuelve las versiones del firmware:

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

b. Mostrar la versión del controlador de la bandeja de entrada:

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

El siguiente ejemplo muestra una versión del controlador:

```
0:14.4.0.6
```

Para obtener la lista actual de versiones de firmware y controladores de adaptador compatibles, consulte la ["Herramienta de matriz de interoperabilidad"](#).

3. Compruebe que `lpfc_enable_fc4_type` se establece en 3:

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

4. Compruebe que puede ver los puertos de iniciador:

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

Deberías ver una salida similar a:

```
0x2100f4c7aa0cd7c2  
0x2100f4c7aa0cd7c3
```

5. Compruebe que los puertos de iniciador estén en línea:

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

Debe ver la siguiente salida:

```
Online  
Online
```

6. Compruebe que los puertos de iniciador NVMe/FC estén habilitados y que los puertos de destino estén visibles:

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

Muestra el ejemplo

```
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

Configure NVMe/FC para un adaptador Marvell/QLogic.

Pasos

1. Verifique que esté utilizando el controlador del adaptador y las versiones de firmware compatibles:

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

El siguiente ejemplo muestra las versiones del controlador y 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. Compruebe que `ql2xnvmeenable` está configurado. Esto permite que el adaptador Marvell funcione como iniciador NVMe/FC:

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

La salida esperada es 1.

NVMe/TCP

El protocolo NVMe/TCP no admite la operación de conexión automática. En su lugar, puede descubrir los subsistemas y espacios de nombres NVMe/TCP realizando la prueba NVMe/TCP. `connect` o `connect-all` operaciones manualmente.

Pasos

1. Verifique que el puerto iniciador pueda obtener los datos de la página de registro de descubrimiento a través de los LIF NVMe/TCP compatibles:

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

Muestra el ejemplo

```
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. Verifique que las otras combinaciones de LIF de iniciador-destino NVMe/TCP puedan recuperar correctamente los datos de la página del registro de descubrimiento:

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

Muestra el ejemplo

```

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. Ejecute el `nvme connect-all` Comando en todos los LIF objetivo iniciador NVMe/TCP admitidos entre los nodos:

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

Muestra el ejemplo

```
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 partir de Rocky Linux 9.4, la configuración para NVMe/TCP `ctrl_loss_tmo timeout` se establece automáticamente en "apagado". Como resultado:

- No hay límites en el número de reintentos (reintento indefinido).
- No es necesario configurar manualmente un elemento específico. `ctrl_loss_tmo timeout` Duración al utilizar el `nvme connect` o `nvme connect-all` comandos (opción `-l`).
- Los controladores NVMe/TCP no experimentan tiempos de espera en caso de una falla de ruta y permanecen conectados indefinidamente.

Paso 4: Opcionalmente, habilite 1 MB de E/S para NVMe/FC

ONTAP informa un tamaño máximo de transferencia de datos (MDTS) de 8 en los datos del controlador de identificación. Esto significa que el tamaño máximo de solicitud de E/S puede ser de hasta 1 MB. Para emitir solicitudes de E/S de tamaño 1 MB para un host Broadcom NVMe/FC, debe aumentar el `lpfc` valor de la `lpfc_sg_seg_cnt` parámetro a 256 desde el valor predeterminado de 64.



Estos pasos no se aplican a los hosts Qlogic NVMe/FC.

Pasos

1. Defina el `lpfc_sg_seg_cnt` parámetro en 256:

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

Debería ver un resultado similar al siguiente ejemplo:

```
options lpfc lpfc_sg_seg_cnt=256
```

2. Ejecute `dracut -f` el comando y reinicie el host.
3. Compruebe que el valor de `lpfc_sg_seg_cnt` es 256:

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

Paso 5: Verificar los servicios de arranque NVMe

El `nvme-fc-boot-connections.service` y `nvmmf-autoconnect.service` Servicios de arranque incluidos en NVMe/FC `nvme-cli` Los paquetes se habilitan automáticamente cuando se inicia el sistema.

Una vez finalizado el arranque, verifique que `nvme-fc-boot-connections.service` y `nvmmf-autoconnect.service` Los servicios de arranque están habilitados.

Pasos

1. Compruebe que `nvmmf-autoconnect.service` está activado:

```
systemctl status nvmmf-autoconnect.service
```

Muestra el resultado de ejemplo

```
nvmmf-autoconnect.service - Connect NVMe-oF subsystems automatically
during boot
   Loaded: loaded (/usr/lib/systemd/system/nvmmf-
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]: nvmmf-
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. Compruebe que `nvme-fc-boot-connections.service` está activado:

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

Muestra el resultado de ejemplo

```
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.
```

Paso 6: Verificar la configuración de rutas múltiples

Verifique que el estado de multivía de NVMe en kernel, el estado de ANA y los espacios de nombres de ONTAP sean correctos para la configuración de NVMe-oF.

Pasos

1. Compruebe que la multivía NVMe en kernel esté habilitada:

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

Debe ver la siguiente salida:

```
Y
```

2. Compruebe que la configuración NVMe-oF adecuada (como, por ejemplo, el modelo configurado en la controladora NetApp ONTAP y la política de balanceo de carga establecida en round-robin) en los respectivos espacios de nombres de ONTAP se reflejen correctamente en el host:

- a. Mostrar los subsistemas:

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

Debe ver la siguiente salida:

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

b. Mostrar la política:

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

Debe ver la siguiente salida:

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

3. Verifique que los espacios de nombres se hayan creado y detectado correctamente en el host:

```
nvme list
```

Muestra el ejemplo

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. Compruebe que el estado de la controladora de cada ruta sea activo y que tenga el estado de ANA correcto:

NVMe/FC

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

Muestra el ejemplo

```
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
```

Muestra el ejemplo

```
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. Confirmar que el complemento de NetApp muestra los valores correctos para cada dispositivo de espacio de nombres ONTAP:

Columna

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

Muestra el ejemplo

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
```

Muestra el ejemplo

```
{
  "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
    }
  ]
}
```

Paso 7: Configurar la autenticación segura en banda

La autenticación segura en banda es compatible con NVMe/TCP entre un host Rocky Linux 10.x y un controlador ONTAP .

Cada host o controlador debe estar asociado con un `DH-HMAC-CHAP` Clave para configurar la autenticación segura. A `DH-HMAC-CHAP` La clave es una combinación del NQN del host o controlador NVMe y un secreto de autenticación configurado por el administrador. Para autenticar su par, un host o una controladora NVMe deben reconocer la clave asociada con el par.

Pasos

Configure la autenticación segura en banda mediante la CLI o un archivo JSON de configuración. Si necesita especificar diferentes claves `dhchap` para diferentes subsistemas, debe utilizar un archivo JSON de configuración.

CLI

Configure la autenticación segura en banda mediante la CLI.

1. Obtenga el NQN del host:

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

2. Genere la clave dhchap para el host Rocky Linux 10.x.

La siguiente salida describe el `gen-dhchap-key` parámetros 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

En el siguiente ejemplo, se genera una clave dhchap aleatoria con HMAC establecido en 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. En la controladora ONTAP, añada el host y especifique ambas claves 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 admite dos tipos de métodos de autenticación: Unidireccional y bidireccional. En el host, conéctese a la controladora ONTAP y especifique claves dhchap según el método de autenticación elegido:

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

5. Valide el `nvme connect authentication` comando mediante la verificación de las claves dhchap de host y controladora:

- a. Verifique las claves dhchap del host:

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

Mostrar ejemplo de salida para una configuración unidireccional

```
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. Compruebe las claves dhchap del controlador:

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

Mostrar ejemplo de salida para una configuración bidireccional

```
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

Cuando hay varios subsistemas NVMe disponibles en el controlador ONTAP , puede utilizar el `/etc/nvme/config.json` archivo con el `nvme connect-all` dominio.

Utilice el `-o` Opción para generar el archivo JSON. Consulte las páginas del manual de NVMe `connect-all` para obtener más opciones de sintaxis.

1. Configurar el archivo JSON.



En el siguiente ejemplo, `dhchap_key` corresponde a `dhchap_secret` y `dhchap_ctrl_key` corresponde a `dhchap_ctrl_secret`.

Muestra el ejemplo

```
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. Conéctese a la controladora ONTAP mediante el archivo JSON de configuración:

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

Muestra el ejemplo

```
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. Verifique que los secretos dhchap se hayan habilitado para los controladores respectivos de cada subsistema.

- a. Verifique las claves dhchap del host:

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

El siguiente ejemplo muestra una clave dhchap:

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

- b. Compruebe las claves dhchap del controlador:

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

Debería ver un resultado similar al siguiente ejemplo:

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

Paso 8: Revise los problemas conocidos

No hay problemas conocidos.

Configurar Rocky Linux 9.x para NVMe-oF con almacenamiento ONTAP

Los hosts Rocky Linux admiten los protocolos NVMe sobre canal de fibra (NVMe/FC) y NVMe sobre TCP (NVMe/TCP) con acceso asimétrico al espacio de nombres (ANA). ANA proporciona una funcionalidad de múltiples rutas equivalente al acceso a unidad lógica asimétrica (ALUA) en entornos iSCSI y FCP.

Aprenda a configurar hosts NVMe over Fabrics (NVMe-oF) para Rocky Linux 9.x. Para obtener más información sobre asistencia y funciones, consulte "[Compatibilidad y características de Rocky Linux ONTAP](#)".

NVMe-oF con Rocky Linux 9.x tiene las siguientes limitaciones conocidas:

- El `nvme disconnect-all` El comando desconecta los sistemas de archivos raíz y de datos y podría provocar inestabilidad del sistema. No emita esto en sistemas que arrancan desde SAN a través de espacios de nombres NVMe-TCP o NVMe-FC.

Paso 1: Opcionalmente, habilite el arranque SAN

Puede configurar su host para utilizar el arranque SAN para simplificar la implementación y mejorar la escalabilidad. Utilice el "[Herramienta de matriz de interoperabilidad](#)" para verificar que su sistema operativo Linux, el adaptador de bus de host (HBA), el firmware del HBA, el BIOS de arranque del HBA y la versión de ONTAP admitan el arranque SAN.

Pasos

1. "[Cree un espacio de nombres NVMe y asígnelo al host](#)".
2. Habilite el arranque SAN en el BIOS del servidor para los puertos a los que está asignado el espacio de nombres de arranque SAN.

Para obtener información acerca de cómo activar el BIOS HBA, consulte la documentación específica de su proveedor.

3. Reinicie el host y verifique que el sistema operativo esté funcionando.

Paso 2: Instale el software Rocky Linux y NVMe y verifique su configuración

Para configurar su host para NVMe-oF, debe instalar los paquetes de software de host y NVMe, habilitar rutas múltiples y verificar la configuración NQN de su host.

Pasos

1. Instalar Rocky Linux 9.x en el servidor. Una vez completada la instalación, verifique que esté ejecutando el kernel Rocky Linux 9.x requerido:

```
uname -r
```

Ejemplo de versión del kernel de Rocky Linux:

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

2. Instale el `nvme-cli` paquete:

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

El siguiente ejemplo muestra una versión del paquete `nvme-cli`:

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

3. Instale el `libnvme` paquete:

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

El siguiente ejemplo muestra un `libnvme` versión del paquete:

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

4. En el host Rocky Linux, verifique la cadena `hostnqn` en `/etc/nvme/hostnqn` :

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

El siguiente ejemplo muestra un `hostnqn` versión:

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

5. En el sistema ONTAP , verifique que `hostnqn` La cadena coincide con la `hostnqn` cadena para el subsistema correspondiente en la matriz ONTAP :

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

Muestra el ejemplo

```
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.
```



Si la `hostnqn` las cadenas no coinciden, utilice `vserver modify` comando para actualizar el `hostnqn` En el subsistema de cabina ONTAP correspondiente de que coincida con `hostnqn` cadena desde `/etc/nvme/hostnqn` en el host.

Paso 3: Configurar NVMe/FC y NVMe/TCP

Configure NVMe/FC con adaptadores Broadcom/Emulex o Marvell/QLogic, o configure NVMe/TCP mediante operaciones de descubrimiento y conexión manuales.

NVMe/FC - Broadcom/Emulex

Configuración de NVMe/FC para un adaptador Broadcom/Emulex.

Pasos

1. Compruebe que está utilizando el modelo de adaptador admitido:

a. Mostrar los nombres de los modelos:

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

Debe ver la siguiente salida:

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

b. Mostrar las descripciones del modelo:

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

Debería ver un resultado similar al siguiente ejemplo:

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

2. Compruebe que está utilizando la Broadcom recomendada lpfc firmware y controlador de bandeja de entrada:

a. Mostrar la versión del firmware:

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

El comando devuelve las versiones del firmware:

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

b. Mostrar la versión del controlador de la bandeja de entrada:

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

El siguiente ejemplo muestra una versión del controlador:

```
0:14.4.0.6
```

Para obtener la lista actual de versiones de firmware y controladores de adaptador compatibles, consulte la "[Herramienta de matriz de interoperabilidad](#)".

3. Compruebe que `lpfc_enable_fc4_type` se establece en 3:

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

4. Compruebe que puede ver los puertos de iniciador:

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

El siguiente ejemplo muestra las identidades del puerto:

```
0x2100f4c7aa0cd7c2  
0x2100f4c7aa0cd7c3
```

5. Compruebe que los puertos de iniciador estén en línea:

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

Debe ver la siguiente salida:

```
Online  
Online
```

6. Compruebe que los puertos de iniciador NVMe/FC estén habilitados y que los puertos de destino estén visibles:

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

Muestra el ejemplo

```
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  Cmpl: 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 Cmpl: 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  Cmpl: 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 Cmpl: xb 000035cc Err 000fcfc0

```

NVMe/FC - Marvell/QLogic

Configure NVMe/FC para un adaptador Marvell/QLogic.

Pasos

1. Verifique que esté utilizando el controlador del adaptador y las versiones de firmware compatibles:

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

El siguiente ejemplo muestra las versiones del controlador y 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. Compruebe que `ql2xnvmeenable` está configurado. Esto permite que el adaptador Marvell funcione como iniciador NVMe/FC:

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

La salida esperada es 1.

NVMe/TCP

El protocolo NVMe/TCP no admite la operación de conexión automática. Debe realizar manualmente las operaciones de conexión NVMe/TCP o de conexión total para descubrir los subsistemas y espacios de nombres NVMe/TCP.

Pasos

1. Verifique que el puerto iniciador pueda obtener los datos de la página de registro de descubrimiento a través de los LIF NVMe/TCP compatibles:

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

Muestra el ejemplo

```
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. Verifique que las otras combinaciones de LIF de iniciador-destino NVMe/TCP puedan obtener correctamente los datos de la página de registro de descubrimiento:

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

Muestra el ejemplo

```
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. Ejecute el `nvme connect-all` Comando en todos los LIF objetivo iniciador NVMe/TCP admitidos entre los nodos:

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

Muestra el ejemplo

```
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 partir de Rocky Linux 9.4, la configuración para NVMe/TCP `ctrl_loss_tmo` timeout se establece automáticamente en "apagado". Como resultado:

- No hay límites en el número de reintentos (reintento indefinido).
- No es necesario configurar manualmente un elemento específico. `ctrl_loss_tmo` timeout Duración al utilizar el `nvme connect` o `nvme connect-all` comandos (opción `-l`).
- Los controladores NVMe/TCP no experimentan tiempos de espera en caso de una falla de ruta y permanecen conectados indefinidamente.

Paso 4: Opcionalmente, habilite 1 MB de E/S para NVMe/FC

ONTAP informa un tamaño máximo de transferencia de datos (MDTS) de 8 en los datos del controlador de identificación. Esto significa que el tamaño máximo de solicitud de E/S puede ser de hasta 1 MB. Para emitir solicitudes de E/S de tamaño 1 MB para un host Broadcom NVMe/FC, debe aumentar el `lpfc` valor de la `lpfc_sg_seg_cnt` parámetro a 256 desde el valor predeterminado de 64.



Estos pasos no se aplican a los hosts Qlogic NVMe/FC.

Pasos

1. Defina el `lpfc_sg_seg_cnt` parámetro en 256:

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

Debería ver un resultado similar al siguiente ejemplo:

```
options lpfc lpfc_sg_seg_cnt=256
```

2. Ejecute `dracut -f` el comando y reinicie el host.
3. Compruebe que el valor de `lpfc_sg_seg_cnt` es 256:

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

Paso 5: Verificar los servicios de arranque NVMe

El `nvme-fc-boot-connections.service` y `nvmmf-autoconnect.service` Servicios de arranque incluidos en NVMe/FC `nvme-cli` Los paquetes se habilitan automáticamente cuando se inicia el sistema.

Una vez finalizado el arranque, verifique que `nvme-fc-boot-connections.service` y `nvmmf-autoconnect.service` Los servicios de arranque están habilitados.

Pasos

1. Compruebe que `nvmmf-autoconnect.service` está activado:

```
systemctl status nvmmf-autoconnect.service
```

Muestra el resultado de ejemplo

```
nvmmf-autoconnect.service - Connect NVMe-oF subsystems automatically
during boot
   Loaded: loaded (/usr/lib/systemd/system/nvmmf-
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]: nvmmf-
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. Compruebe que `nvme-fc-boot-connections.service` está activado:

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

Muestra el resultado de ejemplo

```
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.
```

Paso 6: Verificar la configuración de rutas múltiples

Verifique que el estado de multivía de NVMe en kernel, el estado de ANA y los espacios de nombres de ONTAP sean correctos para la configuración de NVMe-oF.

Pasos

1. Compruebe que la multivía NVMe en kernel esté habilitada:

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

Debe ver la siguiente salida:

```
Y
```

2. Compruebe que la configuración NVMe-oF adecuada (como, por ejemplo, el modelo configurado en la controladora NetApp ONTAP y la política de balanceo de carga establecida en round-robin) en los respectivos espacios de nombres de ONTAP se reflejen correctamente en el host:

- a. Mostrar los subsistemas:

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

Debe ver la siguiente salida:

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

b. Mostrar la política:

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

Debe ver la siguiente salida:

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

3. Verifique que los espacios de nombres se hayan creado y detectado correctamente en el host:

```
nvme list
```

Muestra el ejemplo

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. Compruebe que el estado de la controladora de cada ruta sea activo y que tenga el estado de ANA correcto:

NVMe/FC

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

Muestra el ejemplo

```
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
```

Muestra el ejemplo

```
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. Confirmar que el complemento de NetApp muestra los valores correctos para cada dispositivo de espacio de nombres ONTAP:

Columna

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

Muestra el ejemplo

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
```

Muestra el ejemplo

```
{
  "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
    },
  ]
}
```

Paso 7: Configurar la autenticación segura en banda

La autenticación segura en banda es compatible con NVMe/TCP entre un host Rocky Linux 9x y un controlador ONTAP .

Cada host o controlador debe estar asociado con un `DH-HMAC-CHAP` Clave para configurar la autenticación segura. A `DH-HMAC-CHAP` La clave es una combinación del NQN del host o controlador NVMe y un secreto de autenticación configurado por el administrador. Para autenticar su par, un host o una controladora NVMe deben reconocer la clave asociada con el par.

Pasos

Configure la autenticación segura en banda mediante la CLI o un archivo JSON de configuración. Si necesita especificar diferentes claves `dhchap` para diferentes subsistemas, debe utilizar un archivo JSON de configuración.

CLI

Configure la autenticación segura en banda mediante la CLI.

1. Obtenga el NQN del host:

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

2. Genere la clave dhchap para el host Rocky Linux 9.x.

La siguiente salida describe el `gen-dhchap-key` parámetros 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

En el siguiente ejemplo, se genera una clave dhchap aleatoria con HMAC establecido en 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. En la controladora ONTAP, añada el host y especifique ambas claves 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 admite dos tipos de métodos de autenticación: Unidireccional y bidireccional. En el host, conéctese a la controladora ONTAP y especifique claves dhchap según el método de autenticación elegido:

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

5. Valide el `nvme connect authentication` comando mediante la verificación de las claves dhchap de host y controladora:

- a. Verifique las claves dhchap del host:

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

Mostrar ejemplo de salida para una configuración unidireccional

```
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. Compruebe las claves dhchap del controlador:

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

Mostrar ejemplo de salida para una configuración bidireccional

```
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

Cuando hay varios subsistemas NVMe disponibles en el controlador ONTAP , puede utilizar el `/etc/nvme/config.json` archivo con el `nvme connect-all` dominio.

Utilice el `-o` Opción para generar el archivo JSON. Consulte las páginas del manual de NVMe connect-all para obtener más opciones de sintaxis.

1. Configurar el archivo JSON.



En el siguiente ejemplo, `dhchap_key` corresponde a `dhchap_secret` y `dhchap_ctrl_key` corresponde a `dhchap_ctrl_secret`.

Muestra el ejemplo

```
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. Conéctese a la controladora ONTAP mediante el archivo JSON de configuración:

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

Muestra el ejemplo

```
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. Verifique que se hayan activado los secretos dhchap para las respectivas controladoras de cada subsistema:

a. Verifique las claves dhchap del host:

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

El siguiente ejemplo muestra una clave dhchap:

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

b. Compruebe las claves dhchap del controlador:

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

Debería ver un resultado similar al siguiente ejemplo:

```
DHHC-  
1:03:wSpuuKbBHTzC0W9JZxMBsYd9JFV8Si9aDh22k2BR/4m852vH7KGlrJeMpzhmyjD  
W0o0PJJM6yZsTeEpGkDHMHQ255+g=:
```

Paso 8: Revise los problemas conocidos

No hay problemas conocidos.

Configurar Rocky Linux 8.x para NVMe-oF con almacenamiento ONTAP

Los hosts Rocky Linux admiten los protocolos NVMe sobre canal de fibra (NVMe/FC) y NVMe sobre TCP (NVMe/TCP) con acceso asimétrico al espacio de nombres (ANA). ANA proporciona una funcionalidad de múltiples rutas equivalente al acceso a unidad lógica asimétrica (ALUA) en entornos iSCSI y FCP.

Aprenda a configurar hosts NVMe over Fabrics (NVMe-oF) para Rocky Linux 8.x. Para obtener más información sobre asistencia y funciones, consulte ["Compatibilidad y características de Rocky Linux ONTAP"](#).

NVMe-oF con Rocky Linux 8.x tiene las siguientes limitaciones conocidas:

- Actualmente no se admite el arranque SAN mediante el protocolo NVMe-oF.
- La multiruta NVMe dentro del kernel está deshabilitada de manera predeterminada en los hosts NVMe-oF en Rocky Linux 8.x; debe habilitarla manualmente.
- NVMe/TCP está disponible como una versión preliminar de tecnología debido a problemas conocidos.

Paso 1: Instale el software Rocky Linux y NVMe y verifique su configuración

Para configurar su host para NVMe-oF, debe instalar los paquetes de software de host y NVMe, habilitar rutas múltiples y verificar la configuración NQN de su host.

Pasos

1. Instalar Rocky Linux 8.x en el servidor. Una vez completada la instalación, verifique que esté ejecutando el kernel Rocky Linux 8.x requerido:

```
uname -r
```

Ejemplo de versión del kernel de Rocky Linux:

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

2. Instale el `nvme-cli` paquete:

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

El siguiente ejemplo muestra una versión del paquete nvme-cli:

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

3. Instale el libnvme paquete:

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

El siguiente ejemplo muestra un libnvme versión del paquete:

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

4. En el host Rocky Linux, verifique la cadena hostnqn en /etc/nvme/hostnqn :

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

El siguiente ejemplo muestra un hostnqn versión:

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

5. En el sistema ONTAP , verifique que hostnqn La cadena coincide con la hostnqn cadena para el subsistema correspondiente en la matriz ONTAP :

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

Muestra el ejemplo

```
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.
```



Si la `hostnqn` las cadenas no coinciden, utilice `vserver modify` comando para actualizar el `hostnqn` En el subsistema de cabina ONTAP correspondiente de que coincida con `hostnqn` cadena desde `/etc/nvme/hostnqn` en el host.

Paso 2: Configurar NVMe/FC y NVMe/TCP

Configure NVMe/FC con adaptadores Broadcom/Emulex o Marvell/QLogic, o configure NVMe/TCP mediante operaciones de descubrimiento y conexión manuales.

NVMe/FC - Broadcom/Emulex

Configuración de NVMe/FC para un adaptador Broadcom/Emulex.

Pasos

1. Compruebe que está utilizando el modelo de adaptador admitido:

a. Mostrar los nombres de los modelos:

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

Debe ver la siguiente salida:

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

b. Mostrar las descripciones del modelo:

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

Debería ver un resultado similar al siguiente ejemplo:

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

2. Compruebe que está utilizando la Broadcom recomendada lpfc firmware y controlador de bandeja de entrada:

a. Mostrar la versión del firmware:

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

El comando devuelve las versiones del firmware:

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

b. Mostrar la versión del controlador de la bandeja de entrada:

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

El siguiente ejemplo muestra una versión del controlador:

```
0:14.4.0.2
```

Para obtener la lista actual de versiones de firmware y controladores de adaptador compatibles, consulte la "[Herramienta de matriz de interoperabilidad](#)".

3. Compruebe que la salida esperada de `lpfc_enable_fc4_type` está definida en 3:

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

4. Compruebe que puede ver los puertos de iniciador:

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

El siguiente ejemplo muestra las identidades del puerto:

```
0x100000109bf044b1  
0x100000109bf044b2
```

5. Compruebe que los puertos de iniciador estén en línea:

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

Debe ver la siguiente salida:

```
Online  
Online
```

6. Compruebe que los puertos de iniciador NVMe/FC estén habilitados y que los puertos de destino estén visibles:

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

Muestra el ejemplo

```
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

Configure NVMe/FC para un adaptador Marvell/QLogic.

Pasos

1. Compruebe que está ejecutando las versiones de firmware y controlador del adaptador compatibles:

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

El siguiente ejemplo muestra las versiones del controlador y 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. Compruebe que `ql2xnvmeenable` está configurado. Esto permite que el adaptador Marvell funcione como iniciador NVMe/FC:

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

La salida esperada es 1.

NVMe/TCP

El protocolo NVMe/TCP no admite la operación de conexión automática. En su lugar, puede descubrir los subsistemas y espacios de nombres NVMe/TCP realizando la prueba NVMe/TCP. `connect` o `connect-all` operaciones manualmente.

Pasos

1. Verifique que el puerto iniciador pueda obtener los datos de la página de registro de descubrimiento a través de los LIF NVMe/TCP compatibles:

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

Muestra el ejemplo

```
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. Compruebe que las demás combinaciones de LIF iniciador-destino NVMe/TCP puedan recuperar correctamente los datos de la página de registro de detección:

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

Muestra el ejemplo

```
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. Ejecute el `nvme connect-all` Comando en todos los LIF objetivo iniciador NVMe/TCP admitidos entre los nodos:

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

Muestra el ejemplo

```
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
```

Paso 3: Opcionalmente, habilite 1 MB de E/S para NVMe/FC

Puede habilitar solicitudes de E/S de 1 MB para NVMe/FC configurado con un adaptador Broadcom. ONTAP informa un tamaño máximo de transferencia de datos (MDTS) de 8 en los datos de identificación del controlador. Esto significa que el tamaño máximo de solicitud de E/S puede ser de hasta 1MB TB. Para emitir solicitudes de E/S de 1 MB, debe aumentar el valor `lpfc` de `lpfc_sg_seg_cnt` parámetro a 256 desde el valor predeterminado de 64.



Estos pasos no se aplican a los hosts Qlogic NVMe/FC.

Pasos

1. Defina el `lpfc_sg_seg_cnt` parámetro en 256:

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

```
options lpfc lpfc_sg_seg_cnt=256
```

2. Ejecute `dracut -f` el comando y reinicie el host.
3. Compruebe que el valor de `lpfc_sg_seg_cnt` es 256:

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

Paso 4: Verificar la configuración de rutas múltiples

Verifique que el estado de multivía de NVMe en kernel, el estado de ANA y los espacios de nombres de ONTAP sean correctos para la configuración de NVMe-oF.

Pasos

1. Compruebe que la multivía NVMe en kernel esté habilitada:

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

Debe ver la siguiente salida:

```
Y
```

2. Compruebe que la configuración NVMe-oF adecuada (como, por ejemplo, el modelo configurado en la controladora NetApp ONTAP y la política de balanceo de carga establecida en round-robin) en los respectivos espacios de nombres de ONTAP se reflejen correctamente en el host:

- a. Mostrar los subsistemas:

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

Debe ver la siguiente salida:

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

- b. Mostrar la política:

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

Debe ver la siguiente salida:

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

3. Verifique que los espacios de nombres se hayan creado y detectado correctamente en el host:

```
nvme list
```

Muestra el ejemplo

```
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. Compruebe que el estado de la controladora de cada ruta sea activo y que tenga el estado de ANA correcto:

NVMe/FC

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

Muestra el ejemplo

```
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
```

Muestra el ejemplo

```
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. Confirmar que el complemento de NetApp muestra los valores correctos para cada dispositivo de espacio de nombres ONTAP:

Columna

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

Muestra el ejemplo

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
```

Muestra el ejemplo

```
{
  "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
    },
  ]
}
```

Paso 5: Revisar los problemas conocidos

Estos son los problemas conocidos:

ID de error de NetApp	Título	Descripción
"1479047"	Los hosts NVMe-oF de Rocky Linux 8.x crean controladores de descubrimiento persistentes duplicados	En los hosts NVMe-oF, puede usar el comando "nvme discover -p" para crear controladores de descubrimiento persistentes (PDC). Sin embargo, si está ejecutando Rocky Linux 8.x en un host NVMe-oF, se crea un PDC duplicado cada vez que se ejecuta "nvme discover -p". Cuando se utiliza este comando, solo se debe crear un PDC por combinación iniciador-destino. Sin embargo, si está ejecutando Rocky Linux 8.x en un host NVMe-oF, se crea un PDC duplicado cada vez que se ejecuta "nvme discover -p". Esto genera un uso innecesario de recursos tanto en el host como en el destino.

Información de copyright

Copyright © 2026 NetApp, Inc. Todos los derechos reservados. Imprimido en EE. UU. No se puede reproducir este documento protegido por copyright ni parte del mismo de ninguna forma ni por ningún medio (gráfico, electrónico o mecánico, incluidas fotocopias, grabaciones o almacenamiento en un sistema de recuperación electrónico) sin la autorización previa y por escrito del propietario del copyright.

El software derivado del material de NetApp con copyright está sujeto a la siguiente licencia y exención de responsabilidad:

ESTE SOFTWARE LO PROPORCIONA NETAPP «TAL CUAL» Y SIN NINGUNA GARANTÍA EXPRESA O IMPLÍCITA, INCLUYENDO, SIN LIMITAR, LAS GARANTÍAS IMPLÍCITAS DE COMERCIALIZACIÓN O IDONEIDAD PARA UN FIN CONCRETO, CUYA RESPONSABILIDAD QUEDA EXIMIDA POR EL PRESENTE DOCUMENTO. EN NINGÚN CASO NETAPP SERÁ RESPONSABLE DE NINGÚN DAÑO DIRECTO, INDIRECTO, ESPECIAL, EJEMPLAR O RESULTANTE (INCLUYENDO, ENTRE OTROS, LA OBTENCIÓN DE BIENES O SERVICIOS SUSTITUTIVOS, PÉRDIDA DE USO, DE DATOS O DE BENEFICIOS, O INTERRUPCIÓN DE LA ACTIVIDAD EMPRESARIAL) CUALQUIERA SEA EL MODO EN EL QUE SE PRODUJERON Y LA TEORÍA DE RESPONSABILIDAD QUE SE APLIQUE, YA SEA EN CONTRATO, RESPONSABILIDAD OBJETIVA O AGRAVIO (INCLUIDA LA NEGLIGENCIA U OTRO TIPO), QUE SURJAN DE ALGÚN MODO DEL USO DE ESTE SOFTWARE, INCLUSO SI HUBIEREN SIDO ADVERTIDOS DE LA POSIBILIDAD DE TALES DAÑOS.

NetApp se reserva el derecho de modificar cualquiera de los productos aquí descritos en cualquier momento y sin aviso previo. NetApp no asume ningún tipo de responsabilidad que surja del uso de los productos aquí descritos, excepto aquello expresamente acordado por escrito por parte de NetApp. El uso o adquisición de este producto no lleva implícita ninguna licencia con derechos de patente, de marcas comerciales o cualquier otro derecho de propiedad intelectual de NetApp.

Es posible que el producto que se describe en este manual esté protegido por una o más patentes de EE. UU., patentes extranjeras o solicitudes pendientes.

LEYENDA DE DERECHOS LIMITADOS: el uso, la copia o la divulgación por parte del gobierno están sujetos a las restricciones establecidas en el subpárrafo (b)(3) de los derechos de datos técnicos y productos no comerciales de DFARS 252.227-7013 (FEB de 2014) y FAR 52.227-19 (DIC de 2007).

Los datos aquí contenidos pertenecen a un producto comercial o servicio comercial (como se define en FAR 2.101) y son propiedad de NetApp, Inc. Todos los datos técnicos y el software informático de NetApp que se proporcionan en este Acuerdo tienen una naturaleza comercial y se han desarrollado exclusivamente con fondos privados. El Gobierno de EE. UU. tiene una licencia limitada, irrevocable, no exclusiva, no transferible, no sublicenciable y de alcance mundial para utilizar los Datos en relación con el contrato del Gobierno de los Estados Unidos bajo el cual se proporcionaron los Datos. Excepto que aquí se disponga lo contrario, los Datos no se pueden utilizar, desvelar, reproducir, modificar, interpretar o mostrar sin la previa aprobación por escrito de NetApp, Inc. Los derechos de licencia del Gobierno de los Estados Unidos de América y su Departamento de Defensa se limitan a los derechos identificados en la cláusula 252.227-7015(b) de la sección DFARS (FEB de 2014).

Información de la marca comercial

NETAPP, el logotipo de NETAPP y las marcas que constan en <http://www.netapp.com/TM> son marcas comerciales de NetApp, Inc. El resto de nombres de empresa y de producto pueden ser marcas comerciales de sus respectivos propietarios.