



SUSE Linux Enterprise Server

ONTAP SAN Host Utilities

NetApp
January 30, 2026

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

Sommaire

SUSE Linux Enterprise Server	1
Découvrez la prise en charge et les fonctionnalités ONTAP pour SUSE Linux Enterprise Server	1
Quelle est la prochaine étape	2
Configurer SUSE Linux Enterprise Server 16 pour NVMe-oF avec le stockage ONTAP	2
Étape 1 : activez éventuellement le démarrage SAN	2
Étape 2 : Installez SUSE Linux Enterprise Server et le logiciel NVMe, puis vérifiez votre configuration.	3
Étape 3 : Configurer NVMe/FC et NVMe/TCP	4
Étape 4 : Vous pouvez éventuellement modifier la politique d'E/S dans les règles udev.	13
Étape 5 : Vous pouvez activer l'E/S à 1 Mo pour NVMe/FC (optionnel).	14
Étape 6 : Vérifier les services de démarrage NVMe	15
Étape 7 : Vérifier la configuration du multipathing	16
Étape 8 : Créer un contrôleur de découverte persistant	21
Étape 9 : Configurer une authentification sécurisée en bande	26
Étape 10 : Configurer la sécurité de la couche de transport	32
Étape 11 : passez en revue les problèmes connus	37
Configurer SUSE Linux Enterprise Server 15 SPx pour NVMe-oF avec stockage ONTAP	37
Étape 1 : activez éventuellement le démarrage SAN	38
Étape 2 : Installez SUSE Linux Enterprise Server et le logiciel NVMe, puis vérifiez votre configuration.	38
Étape 3 : Configurer NVMe/FC et NVMe/TCP	40
Étape 4 : Vous pouvez éventuellement modifier la politique d'E/S dans les règles udev.	48
Étape 5 : Vous pouvez activer l'E/S à 1 Mo pour NVMe/FC (optionnel).	49
Étape 6 : Vérifier les services de démarrage NVMe	50
Étape 7 : Vérifier la configuration du multipathing	51
Étape 8 : Créer un contrôleur de découverte persistant	55
Étape 9 : Configurer une authentification sécurisée en bande	60
Étape 10 : Configurer la sécurité de la couche de transport	66
Étape 11 : passez en revue les problèmes connus	71

SUSE Linux Enterprise Server

Découvrez la prise en charge et les fonctionnalités ONTAP pour SUSE Linux Enterprise Server

Les fonctionnalités prises en charge pour la configuration hôte avec NVMe over Fabrics (NVMe-oF) varient en fonction de votre version d'ONTAP et de SUSE Linux Enterprise Server.

Fonctionnalité	Version hôte du serveur d'entreprise SUSE Linux	Version ONTAP
NVMe/TCP signale tous les chemins optimisés pour les systèmes ASA r2	16	9.16.1 ou version ultérieure
Le chiffrement TLS 1.3 (Transport Layer Security) est pris en charge pour NVMe/TCP.	15 SP6 ou version ultérieure	9.16.1 ou version ultérieure
L'authentification sécurisée en bande est prise en charge via NVMe/TCP entre un hôte RHEL et un contrôleur ONTAP.	15 SP4 ou version ultérieure	9.12.1 ou version ultérieure
Les contrôleurs de découverte persistants (PDC) sont pris en charge à l'aide d'un NQN de découverte unique	15 SP4 ou version ultérieure	9.11.1 ou version ultérieure
NVMe/TCP fournit des espaces de noms utilisant le natif <code>nvme-cli</code> emballeur	15 SP4 ou version ultérieure	9.10.1 ou version ultérieure
Le trafic NVMe et SCSI est pris en charge sur le même hôte grâce à NVMe multipath pour les espaces de noms NVMe-oF et à dm-multipath pour les LUN SCSI.	15 SP1 ou version ultérieure	9.4 ou version ultérieure

ONTAP prend en charge les fonctionnalités hôtes SAN suivantes, quelle que soit la version ONTAP exécutée sur votre système.

Fonctionnalité	Version hôte du serveur d'entreprise SUSE Linux
Le démarrage SAN est activé via le protocole NVMe/FC	15 SP7 ou version ultérieure
Le multipathing NVMe natif est activé par défaut	15 SP1 ou version ultérieure
Le <code>nvme-cli</code> Ce package inclut des scripts de connexion automatique, éliminant ainsi le besoin de scripts tiers.	15 SP1 ou version ultérieure



Pour plus de détails sur les configurations prises en charge, consultez le "[Matrice d'interopérabilité](#)" .

Quelle est la prochaine étape

Si votre version de SUSE Linux Enterprise Server est ...	Découvrez ...
16	"Configuration de NVMe pour SUSE Linux Enterprise Server 16"
Série 15 SPx	"Configuration de NVMe pour SUSE Linux Enterprise Server 15 SPx"

Informations associées

- ["Découvrez les systèmes ASA r2"](#)
- ["Découvrez comment gérer les protocoles NVMe"](#)

Configurer SUSE Linux Enterprise Server 16 pour NVMe-oF avec le stockage ONTAP

Le serveur hôte SUSE Linux Enterprise Server 16 prend en charge les protocoles NVMe sur Fibre Channel (NVMe/FC) et NVMe sur TCP (NVMe/TCP) avec Asymmetric Namespace Access (ANA). ANA offre une fonctionnalité de multipathing équivalente à l'asymmetric logical unit access (ALUA) dans les environnements iSCSI et FCP.

Découvrez comment configurer les hôtes NVMe over Fabrics (NVMe-oF) pour SUSE Linux Enterprise Server 16. Pour plus d'informations sur l'assistance et les fonctionnalités, consultez "["Prise en charge et fonctionnalités ONTAP"](#)" .

NVMe-oF avec SUSE Linux Enterprise Server 16 présente les limitations connues suivantes :

- Le `nvme disconnect-all` Cette commande déconnecte les systèmes de fichiers racine et de données et peut entraîner une instabilité du système. Ne pas utiliser cette commande sur des systèmes démarrant à partir d'un SAN via des espaces de noms NVMe-TCP ou NVMe-FC.
- La prise en charge de l'utilitaire hôte `sanlun` de NetApp n'est pas disponible pour NVMe-oF. Vous pouvez en revanche utiliser le plug-in NetApp inclus dans le logiciel natif. `nvme-cli` package pour tous les transports NVMe-oF.

Étape 1 : activez éventuellement le démarrage SAN

Vous pouvez configurer votre hôte pour utiliser le démarrage SAN afin de simplifier le déploiement et d'améliorer l'évolutivité. Utilisez le "["Matrice d'interopérabilité"](#)" pour vérifier que votre système d'exploitation Linux, votre adaptateur de bus hôte (HBA), votre micrologiciel HBA, votre BIOS de démarrage HBA et votre version ONTAP prennent en charge le démarrage SAN.

Étapes

1. ["Créez un espace de noms NVMe et mappez-le à l'hôte"](#) .
2. Activez le démarrage SAN dans le BIOS du serveur pour les ports auxquels l'espace de noms de démarrage SAN est mappé.

Pour plus d'informations sur l'activation du BIOS HBA, reportez-vous à la documentation spécifique au fournisseur.

3. Redémarrez l'hôte et vérifiez que le système d'exploitation est opérationnel.

Étape 2 : Installez SUSE Linux Enterprise Server et le logiciel NVMe, puis vérifiez votre configuration.

Pour configurer votre hôte pour NVMe-oF, vous devez installer les packages logiciels hôte et NVMe, activer le multipathing et vérifier la configuration NQN de votre hôte.

Étapes

1. Installez SUSE Linux Enterprise Server 16 sur le serveur. Après l'installation, vérifiez que vous utilisez le noyau SUSE Linux Enterprise Server 16 spécifié :

```
uname -r
```

Exemple de version du noyau SUSE Linux Enterprise Server :

```
6.12.0-160000.6-default
```

2. Installer le `nvme-cli` groupe :

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

L'exemple suivant montre un `nvme-cli` version du paquet :

```
nvme-cli-2.11+29.g35e62868-160000.1.1.x86_64
```

3. Installer le `libnvme` groupe :

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

L'exemple suivant montre un `libnvme` version du paquet :

```
libnvme1-1.11+17.g6d55624d-160000.1.1.x86_64
```

4. Sur l'hôte, vérifiez la chaîne `hostnqn` à `/etc/nvme/hostnqn` :

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

L'exemple suivant montre un hostnqn version:

```
nqn.2014-08.org.nvmeexpress:uuid:d3b581b4-c975-11e6-8425-0894ef31a074
```

5. Sur le système ONTAP , vérifiez que le hostnqn La chaîne correspond à hostnqn chaîne de caractères pour le sous-système correspondant sur la matrice ONTAP :

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

Montrer l'exemple

```
Vserver Subsystem Priority Host NQN
----- -----
-----
vs_coexistence_emulex
    nvme1
        regular    nqn.2014-
08.org.nvmeexpress:uuid:d3b581b4-c975-11e6-8425-0894ef31a074
    nvme10
        regular    nqn.2014-
08.org.nvmeexpress:uuid:d3b581b4-c975-11e6-8425-0894ef31a074
    nvme11
        regular    nqn.2014-
08.org.nvmeexpress:uuid:d3b581b4-c975-11e6-8425-0894ef31a074
    nvme12
        regular    nqn.2014-
08.org.nvmeexpress:uuid:d3b581b4-c975-11e6-8425-0894ef31a074
4 entries were displayed.
```



Si le hostnqn les chaînes ne correspondent pas, utilisez le vserver modify commande pour mettre à jour le hostnqn Chaîne sur le sous-système de matrice ONTAP correspondant à hostnqn chaîne de /etc/nvme/hostnqn sur l'hôte.

Étape 3 : Configurer NVMe/FC et NVMe/TCP

Configurez NVMe/FC avec des adaptateurs Broadcom/Emulex ou Marvell/QLogic, ou configurez NVMe/TCP à l'aide d'opérations de découverte et de connexion manuelles.

NVMe/FC - Broadcom/Emulex

Configuration de NVMe/FC pour une carte FC Broadcom/Emulex

Étapes

1. Vérifiez que vous utilisez le modèle d'adaptateur pris en charge :

- a. Afficher les noms des modèles :

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

Vous devriez voir le résultat suivant :

```
SN37A92079  
SN37A92079
```

- b. Afficher les descriptions des modèles :

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

Vous devriez voir le résultat suivant :

```
Emulex SN37A92079 32Gb 2-Port Fibre Channel Adapter  
Emulex SN37A92079 32Gb 2-Port Fibre Channel Adapter
```

2. Vérifiez que vous utilisez la carte Broadcom recommandée lpfc micrologiciel et pilote de boîte de réception :

- a. Afficher la version du firmware :

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

L'exemple suivant montre les versions du firmware :

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

- b. Afficher la version du pilote de la boîte de réception :

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

L'exemple suivant montre une version de pilote :

```
0:14.4.0.11
```

Pour obtenir la liste actuelle des versions de pilotes et de micrologiciels de carte prises en charge, consultez le "[Matrice d'interopérabilité](#)".

3. Vérifiez que la sortie attendue de `lpfc_enable_fc4_type` est définie sur 3:

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

4. Vérifiez que vous pouvez afficher vos ports initiateurs :

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

Vous devriez voir une sortie similaire à :

```
0x100000109bdacc75  
0x100000109bdacc76
```

5. Vérifiez que vos ports initiateurs sont en ligne :

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

Vous devriez voir le résultat suivant :

```
Online  
Online
```

6. Vérifiez que les ports initiateurs NVMe/FC sont activés et que les ports cibles sont visibles :

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

Affiche un exemple de résultat

```
NVME Initiator Enabled
XRI Dist lpfco Total 6144 IO 5894 ELS 250
NVME LPORT lpfco WWPN x100000109bdacc75 WWNN x200000109bdacc75
DID x060100 ONLINE
NVME RPORT WWPN x2001d039ea951c45 WWNN x2000d039ea951c45
DID x080801 TARGET DISCSRVC ONLINE
NVME RPORT WWPN x2003d039ea951c45 WWNN x2000d039ea951c45
DID x080d01 TARGET DISCSRVC ONLINE
NVME RPORT WWPN x2024d039eab31e9c WWNN x2023d039eab31e9c
DID x020a09 TARGET DISCSRVC ONLINE
NVME RPORT WWPN x2026d039eab31e9c WWNN x2023d039eab31e9c
DID x020a08 TARGET DISCSRVC ONLINE
NVME RPORT WWPN x2003d039ea5cf90 WWNN x2002d039ea5cf90
DID x061b01 TARGET DISCSRVC ONLINE
NVME RPORT WWPN x2012d039ea5cf90 WWNN x2011d039ea5cf90
DID x061b05 TARGET DISCSRVC ONLINE
NVME RPORT WWPN x2005d039ea5cf90 WWNN x2002d039ea5cf90
DID x061201 TARGET DISCSRVC ONLINE
NVME RPORT WWPN x2014d039ea5cf90 WWNN x2011d039ea5cf90
DID x061205 TARGET DISCSRVC ONLINE

NVME Statistics
LS: Xmt 0000017242 Cmpl 0000017242 Abort 00000000
LS XMIT: Err 00000000 CMPL: xb 00000000 Err 00000000
Total FCP Cmpl 000000000378362 Issue 0000000003783c7 OutIO
0000000000000065
    abort 00000409 noxri 00000000 nondlp 0000003a qdepth
00000000 wqerr 00000000 err 00000000
FCP CMPL: xb 00000409 Err 0000040a

NVME Initiator Enabled
XRI Dist lpfcl Total 6144 IO 5894 ELS 250
NVME LPORT lpfcl WWPN x100000109bdacc76 WWNN x200000109bdacc76
DID x062800 ONLINE
NVME RPORT WWPN x2002d039ea951c45 WWNN x2000d039ea951c45
DID x080701 TARGET DISCSRVC ONLINE
NVME RPORT WWPN x2004d039ea951c45 WWNN x2000d039ea951c45
DID x081501 TARGET DISCSRVC ONLINE
NVME RPORT WWPN x2025d039eab31e9c WWNN x2023d039eab31e9c
DID x020913 TARGET DISCSRVC ONLINE
NVME RPORT WWPN x2027d039eab31e9c WWNN x2023d039eab31e9c
DID x020912 TARGET DISCSRVC ONLINE
NVME RPORT WWPN x2006d039ea5cf90 WWNN x2002d039ea5cf90
DID x061401 TARGET DISCSRVC ONLINE
```

```
NVME RPORT      WWPN x2015d039ea5cf90 WWNN x2011d039ea5cf90
DID x061405 TARGET DISCSRVC ONLINE
NVME RPORT      WWPN x2004d039ea5cf90 WWNN x2002d039ea5cf90
DID x061301 TARGET DISCSRVC ONLINE
NVME RPORT      WWPN x2013d039ea5cf90 WWNN x2011d039ea5cf90
DID x061305 TARGET DISCSRVC ONLINE

NVME Statistics
LS: Xmt 0000017428 Cmpl 0000017428 Abort 00000000
LS XMIT: Err 00000000 CMPL: xb 00000000 Err 00000000
Total FCP Cmpl 0000000003443be Issue 00000000034442a OutIO
000000000000006c
          abort 00000491 noxri 00000000 nondlp 00000086 qdepth
00000000 wqerr 00000000 err 00000000
FCP CMPL: xb 00000491 Err 00000494
```

NVMe/FC - Marvell/QLogic

Configuration du NVMe/FC pour un adaptateur Marvell/QLogic

Étapes

1. Vérifiez que vous exécutez les versions du pilote de carte et du micrologiciel prises en charge :

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

L'exemple suivant montre les versions du pilote et du micrologiciel :

QLE2772 FW:v9.15.06 DVR:v10.02.09.400-k-debug
QLE2772 FW:v9.15.06 DVR:v10.02.09.400-k-debug

2. Vérifiez-le ql2xnvmeenable est défini. L'adaptateur Marvell peut ainsi fonctionner en tant qu'initiateur NVMe/FC :

```
cat /sys/module/q1a2xxx/parameters/q12xnvmeeable
```

La sortie attendue est 1.

NVMe/TCP

Le protocole NVMe/TCP ne prend pas en charge l'opération de connexion automatique. Au lieu de cela, vous pouvez découvrir les sous-systèmes et espaces de noms NVMe/TCP en exécutant l'opération NVMe/TCP connect ou connect-all opérations manuellement.

Étapes

1. Vérifiez que le port initiateur peut récupérer les données de la page de journal de découverte sur les LIF NVMe/TCP prises en charge :

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

Affiche un exemple de résultat

```
nvme discover -t tcp -w 192.168.38.20 -a 192.168.38.10
Discovery Log Number of Records 8, Generation counter 42
=====Discovery Log Entry 0=====
trtype: tcp
adrifam: ipv4
subtype: current discovery subsystem
treq: not specified
portid: 4
trsvcid: 8009
subnqn: nqn.1992-
08.com.netapp:sn.f8e2af201b7211f0ac2bd039eab67a95:discovery
traddr: 192.168.211.71
eflags: explicit discovery connections, duplicate discovery
information
sectype: none
=====Discovery Log Entry 1=====
trtype: tcp
adrifam: ipv4
subtype: current discovery subsystem
treq: not specified
portid: 3
trsvcid: 8009
subnqn: nqn.1992-
08.com.netapp:sn.f8e2af201b7211f0ac2bd039eab67a95:discovery
traddr: 192.168.111.71
eflags: explicit discovery connections, duplicate discovery
information
sectype: none
=====Discovery Log Entry 2=====
trtype: tcp
adrifam: ipv4
subtype: current discovery subsystem
treq: not specified
portid: 2
trsvcid: 8009
subnqn: nqn.1992-
08.com.netapp:sn.f8e2af201b7211f0ac2bd039eab67a95:discovery
traddr: 192.168.211.70
eflags: explicit discovery connections, duplicate discovery
information
sectype: none
=====Discovery Log Entry 3=====
trtype: tcp
adrifam: ipv4
```

```
subtype: current discovery subsystem
treq:    not specified
portid:  1
trsvcid: 8009
subnqn:  nqn.1992-
08.com.netapp:sn.f8e2af201b7211f0ac2bd039eab67a95:discovery
traddr:  192.168.111.70
eflags:  explicit discovery connections, duplicate discovery
information
sectype: none
=====Discovery Log Entry 4=====
trtype:  tcp
adrifam:  ipv4
subtype: nvme subsystem
treq:    not specified
portid:  4
trsvcid: 4420
subnqn:  nqn.1992-
08.com.netapp:sn.f8e2af201b7211f0ac2bd039eab67a95:subsystem.samp
le_tcp_sub
traddr:  192.168.211.71
eflags:  none
sectype: none
=====Discovery Log Entry 5=====
trtype:  tcp
adrifam:  ipv4
subtype: nvme subsystem
treq:    not specified
portid:  3
trsvcid: 4420
subnqn:  nqn.1992-
08.com.netapp:sn.f8e2af201b7211f0ac2bd039eab67a95:subsystem.samp
le_tcp_sub
traddr:  192.168.111.71
eflags:  none
sectype: none
=====Discovery Log Entry 6=====
trtype:  tcp
adrifam:  ipv4
subtype: nvme subsystem
treq:    not specified
portid:  2
trsvcid: 4420
subnqn:  nqn.1992-
08.com.netapp:sn.f8e2af201b7211f0ac2bd039eab67a95:subsystem.samp
le_tcp_sub
```

```
traddr: 192.168.211.70
eflags: none
sectype: none
=====Discovery Log Entry 7=====
trtype: tcp
adrfrm: ipv4
subtype: nvme subsystem
treq: not specified
portid: 1
trsvcid: 4420
subnqn: nqn.1992-
08.com.netapp:sn.f8e2af201b7211f0ac2bd039eab67a95:subsystem.sample_tcp_sub
traddr: 192.168.111.70
eflags: none
sectype: none
localhost:~ #
```

2. Vérifier que toutes les autres combinaisons de LIF NVMe/TCP initiator-target peuvent récupérer les données de la page du journal de découverte :

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

Montrer l'exemple

```
nvme discover -t tcp -w 192.168.38.20 -a 192.168.38.10
nvme discover -t tcp -w 192.168.38.20 -a 192.168.38.11
nvme discover -t tcp -w 192.168.39.20 -a 192.168.39.10
nvme discover -t tcp -w 192.168.39.20 -a 192.168.39.11
```

3. Exécutez le `nvme connect-all` Commande sur toutes les LIF cible-initiateur NVMe/TCP prises en charge sur l'ensemble des nœuds :

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

Montrer l'exemple

```
nvme connect-all -t tcp -w 192.168.38.20 -a  
192.168.38.10  
nvme connect-all -t tcp -w 192.168.38.20 -a  
192.168.38.11  
nvme connect-all -t tcp -w 192.168.39.20 -a  
192.168.39.10  
nvme connect-all -t tcp -w 192.168.39.20 -a  
192.168.39.11
```

Le paramètre pour NVMe/TCP `ctrl_loss_tmo timeout` est automatiquement réglé sur « désactivé ». Par conséquent :

- Il n'y a pas de limite au nombre de tentatives (nouvelle tentative indéfinie).
- Vous n'avez pas besoin de configurer manuellement un élément spécifique `ctrl_loss_tmo timeout` durée lors de l'utilisation du `nvme connect` ou `nvme connect-all` commandes (option `-l`).
- Les contrôleurs NVMe/TCP ne subissent pas de dépassement de délai en cas de défaillance d'un chemin et restent connectés indéfiniment.

Étape 4 : Vous pouvez éventuellement modifier la politique d'E/S dans les règles udev.

À partir de SUSE Linux Enterprise Server 16, la iopolicy par défaut pour NVMe-oF est définie sur `queue-depth`. Si vous souhaitez changer la iopolicy en `round-robin`, modifiez le fichier de règles udev comme suit :

Étapes

1. Ouvrez le fichier de règles udev dans un éditeur de texte avec des privilèges root :

```
/usr/lib/udev/rules.d/71-nvme.rules
```

Vous devriez voir le résultat suivant :

```
vi /usr/lib/udev/rules.d/71-nvme.rules
```

2. Recherchez la ligne qui définit la politique d'E/S pour le contrôleur NetApp ONTAP , comme indiqué dans l'exemple de règle suivant :

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

3. Modifiez la règle afin que queue-depth devienne round-robin :

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

4. Rechargez les règles udev et appliquez les modifications :

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

5. Vérifiez la politique d'E/S actuelle de votre sous-système. Remplacez <subsystem>, par exemple, nvme-subsys0 .

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

Vous devriez voir le résultat suivant :

```
round-robin
```

 La nouvelle politique d'E/S s'applique automatiquement aux périphériques NetApp ONTAP Controller correspondants. Vous n'avez pas besoin de redémarrer.

Étape 5 : Vous pouvez activer l'E/S à 1 Mo pour NVMe/FC (optionnel).

ONTAP signale une taille de transfert de données maximale (MDTS) de 8 dans les données du contrôleur d'identification. Cela signifie que la taille maximale de la demande d'E/S peut atteindre 1 Mo. Pour émettre des requêtes d'E/S d'une taille de 1 Mo pour un hôte Broadcom NVMe/FC, vous devez augmenter la `1pfc` valeur de la `1pfc_sg_seg_cnt` paramètre à 256 à partir de la valeur par défaut de 64.

 Ces étapes ne s'appliquent pas aux hôtes NVMe/FC Qlogic.

Étapes

1. Réglez le `1pfc_sg_seg_cnt` paramètre sur 256 :

```
cat /etc/modprobe.d/1pfc.conf
```

Vous devriez voir une sortie similaire à l'exemple suivant :

```
options lpfc lpfc_sg_seg_cnt=256
```

2. Exécutez `dracut -f` la commande et redémarrez l'hôte.
3. Vérifier que la valeur de `lpfc_sg_seg_cnt` est 256 :

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

Étape 6 : Vérifier les services de démarrage NVMe

Le `nvmefc-boot-connections.service` et `nvmf-autoconnect.service` services de démarrage inclus dans NVMe/FC `nvme-cl` les packages sont automatiquement activés au démarrage du système.

Une fois le démarrage terminé, vérifiez que le `nvmefc-boot-connections.service` et `nvmf-autoconnect.service` les services de démarrage sont activés.

Étapes

1. Vérifiez que `nvmf-autoconnect.service` est activé :

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

Affiche un exemple de résultat

```
nvmf-autoconnect.service - Connect NVMe-oF subsystems automatically
during boot
   Loaded: loaded (/usr/lib/systemd/system/nvmf-autoconnect.service;
             enabled; vendor preset: disabled)
   Active: inactive (dead) since Thu 2024-05-25 14:55:00 IST; 11min
             ago
     Process: 2108 ExecStartPre=/sbin/modprobe nvme-fabrics (code=exited,
               status=0/SUCCESS)
     Process: 2114 ExecStart=/usr/sbin/nvme connect-all (code=exited,
               status=0/SUCCESS)
   Main PID: 2114 (code=exited, status=0/SUCCESS)

systemd[1]: Starting Connect NVMe-oF subsystems automatically during
boot...
nvme[2114]: traddr=nn-0x201700a098fd4ca6:pn-0x201800a098fd4ca6 is
already connected
systemd[1]: nvmf-autoconnect.service: Deactivated successfully.
systemd[1]: Finished Connect NVMe-oF subsystems automatically during
boot.
```

2. Vérifiez que `nvmefc-boot-connections.service` est activé :

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

Affiche un exemple de résultat

```
nvmefc-boot-connections.service - Auto-connect to subsystems on FC-NVME devices found during boot
   Loaded: loaded (/usr/lib/systemd/system/nvmefc-boot-connections.service; enabled; vendor preset: enabled)
   Active: inactive (dead) since Thu 2024-05-25 14:55:00 IST; 11min ago
     Main PID: 1647 (code=exited, status=0/SUCCESS)

systemd[1]: Starting Auto-connect to subsystems on FC-NVME devices found during boot...
systemd[1]: nvmefc-boot-connections.service: Succeeded.
systemd[1]: Finished Auto-connect to subsystems on FC-NVME devices found during boot.
```

Étape 7 : Vérifier la configuration du multipathing

Vérifiez que l'état des chemins d'accès multiples NVMe in-kernel, l'état ANA et les namespaces ONTAP sont corrects pour la configuration NVMe-of.

Étapes

1. Vérifiez que le chemin d'accès multiples NVMe intégré au noyau est activé :

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

Vous devriez voir le résultat suivant :

```
Y
```

2. Vérifiez que les paramètres NVMe-oF appropriés (tels que le modèle défini sur NetApp ONTAP Controller et la stratégie d'E/S d'équilibrage de charge définie sur queue-depth) pour les espaces de noms ONTAP respectifs se reflètent correctement sur l'hôte :

- a. Afficher les sous-systèmes :

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

Vous devriez voir le résultat suivant :

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

b. Afficher la politique :

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

Vous devriez voir le résultat suivant :

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

3. Vérifiez que les espaces de noms sont créés et correctement découverts sur l'hôte :

```
nvme list
```

Montrer l'exemple

Node	SN	Model		
/dev/nvme7n1	81Ix2BVuekWcAAAAAAAB	NetApp ONTAP Controller		
Namespace	Usage	Format	FW	Rev
-----	21.47 GB / 21.47 GB		4 KiB + 0 B	FFFFFFFFFF

4. Vérifiez que l'état du contrôleur de chaque chemin est actif et que l'état ANA est correct :

```
nvme list-subsy /dev/<controller_ID>
```



À partir d'ONTAP 9.16.1, NVMe/FC et NVMe/TCP signalent tous les chemins optimisés sur les systèmes ASA r2.

NVMe/FC

L'exemple suivant montre les résultats pour un espace de noms hébergé sur un contrôleur ONTAP à deux nœuds pour les systèmes AFF, FAS et ASA ainsi que le système ASA r2 avec NVMe/FC.

Afficher un exemple de sortie AFF, FAS et ASA

```
nvme-subsy114 - NQN=nqn.1992-
08.com.netapp:sn.9e30b9760a4911f08c87d039eab67a95:subsystem.sles
_161_27
    hostnqn=nqn.2014-
08.org.nvmeexpress:uuid:f6517cae-3133-11e8-bbff-7ed30aef123f
iopolicy=round-robin\
+- nvme114 fc traddr=nn-0x234ed039ea359e4a:pn-
0x2360d039ea359e4a,host_traddr=nn-0x20000090fae0ec88:pn-
0x10000090fae0ec88 live optimized
+- nvme115 fc traddr=nn-0x234ed039ea359e4a:pn-
0x2362d039ea359e4a,host_traddr=nn-0x20000090fae0ec88:pn-
0x10000090fae0ec88 live non-optimized
+- nvme116 fc traddr=nn-0x234ed039ea359e4a:pn-
0x2361d039ea359e4a,host_traddr=nn-0x20000090fae0ec89:pn-
0x10000090fae0ec89 live optimized
+- nvme117 fc traddr=nn-0x234ed039ea359e4a:pn-
0x2363d039ea359e4a,host_traddr=nn-0x20000090fae0ec89:pn-
0x10000090fae0ec89 live non-optimized
```

Afficher un exemple de sortie ASA r2

```
nvme-subsy96 - NQN=nqn.1992-
08.om.netapp:sn.b351b2b6777b11f0b3c2d039ea5cf91:subsystem.nvme2
4
        hostnqn=nqn.2014-
08.org.nvmeexpress:uuid:d3b581b4-c975-11e6-8425-0894ef31a074
\
    +- nvme203 fc traddr=nn-0x2011d039ea5cf90:pn-
0x2015d039ea5cf90,host_traddr=nn-0x200000109bdacc76:pn-
0x100000109bdacc76 live optimized
    +- nvme25 fc traddr=nn-0x2011d039ea5cf90:pn-
0x2014d039ea5cf90,host_traddr=nn-0x200000109bdacc75:pn-
0x100000109bdacc75 live optimized
    +- nvme30 fc traddr=nn-0x2011d039ea5cf90:pn-
0x2012d039ea5cf90,host_traddr=nn-0x200000109bdacc75:pn-
0x100000109bdacc75 live optimized
    +- nvme32 fc traddr=nn-0x2011d039ea5cf90:pn-
0x2013d039ea5cf90,host_traddr=nn-0x200000109bdacc76:pn-
0x100000109bdacc76 live optimized
```

NVMe/TCP

L'exemple suivant montre des sorties affichant un espace de noms hébergé sur un contrôleur ONTAP à deux nœuds pour les systèmes AFF, FAS et ASA ainsi que les systèmes ASA r2 avec NVMe/TCP.

Afficher un exemple de sortie AFF, FAS et ASA

```
nvme-subsy9 - NQN=nqn.1992-
08.com.netapp:sn.9927e165694211f0b4f4d039eab31e9d:subsystem.nvme
10
hostnqn=nqn.2014-08.org.nvmeexpress:uuid:4c4c4544-
0035-5910-804b-b7c04f444d33
\
+- nvme105 tcp
traddr=192.168.39.10,trsvcid=4420,host_traddr=192.168.39.20,src_
addr=192.168.39.20 live optimized
+- nvme153 tcp
traddr=192.168.39.11,trsvcid=4420,host_traddr=192.168.39.20,src_
addr=192.168.39.20 live non-optimized
+- nvme57 tcp
traddr=192.168.38.11,trsvcid=4420,host_traddr=192.168.38.20,src_
addr=192.168.38.20 live non-optimized
+- nvme9 tcp
traddr=192.168.38.10,trsvcid=4420,host_traddr=192.168.38.20,src_
addr=192.168.38.20 live optimized
```

Afficher un exemple de sortie ASA r2

```
nvme-subsy4 - NQN=nqn.1992-
08.com.netapp:sn.17e32b6e8c7f11f09545d039eac03c33:subsystem.Bidi
rectional_DHCP_1_0
hostnqn=nqn.2014-08.org.nvmeexpress:uuid:4c4c4544-
0054-5110-8039-c3c04f523034
\
+- nvme4 tcp
traddr=192.168.20.28,trsvcid=4420,host_traddr=192.168.20.21,src_
addr=192.168.20.21 live optimized
+- nvme5 tcp
traddr=192.168.20.29,trsvcid=4420,host_traddr=192.168.20.21,src_
addr=192.168.20.21 live optimized
+- nvme6 tcp
traddr=192.168.21.28,trsvcid=4420,host_traddr=192.168.21.21,src_
addr=192.168.21.21 live optimized
+- nvme7 tcp
traddr=192.168.21.29,trsvcid=4420,host_traddr=192.168.21.21,src_
addr=192.168.21.21 live optimized
```

5. Vérifier que le plug-in NetApp affiche les valeurs correctes pour chaque périphérique d'espace de noms ONTAP :

Colonne

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

Montrer l'exemple

Device NSID UUID	Vserver	Namespace Path	Size
/dev/nvme0n1 79510f05-7784-11f0-b3c2-d039ea5cf91	vs_coexistence_emulex	ns1	1 21.47GB

JSON

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

Montrer l'exemple

```
{  
  "ONTAPdevices": [  
    {  
      "Device": "/dev/nvme0n1",  
      "Vserver": "vs_coexistence_emulex",  
      "Namespace_Path": "ns1",  
      "NSID": 1,  
      "UUID": "79510f05-7784-11f0-b3c2-d039ea5cf91",  
      "Size": "21.47GB",  
      "LBA_Data_Size": 4096,  
      "Namespace_Size": 5242880  
    }  
  ]  
}
```

Étape 8 : Créer un contrôleur de découverte persistant

Vous pouvez créer un contrôleur de découverte persistant (PDC) pour un hôte SUSE Linux Enterprise Server 16. Un PDC est nécessaire pour détecter automatiquement une opération d'ajout ou de suppression de sous-système NVMe et les changements apportés aux données de la page de journal de découverte.

Étapes

1. Vérifier que les données de la page du journal de découverte sont disponibles et peuvent être récupérées via la combinaison port initiateur et LIF cible :

```
nvme discover -t <trtype> -w <host-traddr> -a <traddr>
```

Affiche un exemple de résultat

```
Discovery Log Number of Records 8, Generation counter 10
=====Discovery Log Entry 0=====
trtype: tcp
adrifam: ipv4
subtype: current discovery subsystem
treq: not specified
portid: 3
trsvcid: 8009
subnqn: nqn.1992-
08.com.netapp:sn.9927e165694211f0b4f4d039eab31e9d:discovery
traddr: 192.168.39.10
eflags: explicit discovery connections, duplicate discovery
information
sectype: none
=====Discovery Log Entry 1=====
trtype: tcp
adrifam: ipv4
subtype: current discovery subsystem
treq: not specified
portid: 1
trsvcid: 8009
subnqn: nqn.1992-
08.com.netapp:sn.9927e165694211f0b4f4d039eab31e9d:discovery
traddr: 192.168.38.10
eflags: explicit discovery connections, duplicate discovery
information
sectype: none
=====Discovery Log Entry 2=====
trtype: tcp
adrifam: ipv4
subtype: current discovery subsystem
treq: not specified
portid: 4
trsvcid: 8009
subnqn: nqn.1992-
08.com.netapp:sn.9927e165694211f0b4f4d039eab31e9d:discovery
traddr: 192.168.39.11
eflags: explicit discovery connections, duplicate discovery
information
sectype: none
=====Discovery Log Entry 3=====
trtype: tcp
adrifam: ipv4
subtype: current discovery subsystem
```

```
treq: not specified
portid: 2
trsvcid: 8009
subnqn: nqn.1992-
08.com.netapp:sn.9927e165694211f0b4f4d039eab31e9d:discovery
traddr: 192.168.38.11
eflags: explicit discovery connections, duplicate discovery
information
sectype: none
=====Discovery Log Entry 4=====
trtype: tcp
adrifam: ipv4
subtype: nvme subsystem
treq: not specified
portid: 3
trsvcid: 4420
subnqn: nqn.1992-
08.com.netapp:sn.9927e165694211f0b4f4d039eab31e9d:subsystem.nvme1
traddr: 192.168.39.10
eflags: none
sectype: none
=====Discovery Log Entry 5=====
trtype: tcp
adrifam: ipv4
subtype: nvme subsystem
treq: not specified
portid: 1
trsvcid: 4420
subnqn: nqn.1992-
08.com.netapp:sn.9927e165694211f0b4f4d039eab31e9d:subsystem.nvme1
traddr: 192.168.38.10
eflags: none
sectype: none
=====Discovery Log Entry 6=====
trtype: tcp
adrifam: ipv4
subtype: nvme subsystem
treq: not specified
portid: 4
trsvcid: 4420
subnqn: nqn.1992-
08.com.netapp:sn.9927e165694211f0b4f4d039eab31e9d:subsystem.nvme1
traddr: 192.168.39.11
eflags: none
sectype: none
=====Discovery Log Entry 7=====
```

```
trtype:    tcp
adrifam:   ipv4
subtype:   nvme subsystem
treq:      not specified
portid:    2
trsvcid:  4420
subnqn:   nqn.1992-
08.com.netapp:sn.9927e165694211f0b4f4d039eab31e9d:subsystem.nvme1
traddr:   192.168.38.11
eflags:   none
sectype:  none
```

2. Créer un PDC pour le sous-système de découverte :

```
nvme discover -t <trtype> -w <host-traddr> -a <traddr> -p
```

Vous devriez voir le résultat suivant :

```
nvme discover -t tcp -w 192.168.39.20 -a 192.168.39.11 -p
```

3. À partir du contrôleur ONTAP, vérifier que le PDC a été créé :

```
vserver nvme show-discovery-controller -instance -vserver <vserver_name>
```

Affiche un exemple de résultat

```
vserver nvme show-discovery-controller -instance -vserver
vs_tcp_sles16
Vserver Name: vs_tcp_sles16
    Controller ID: 0180h
    Discovery Subsystem NQN: nqn.1992-
08.com.netapp:sn.9927e165694211f0b4f4d039eab31e9d:discovery
    Logical Interface: lif3
    Node: A400-12-171
    Host NQN: nqn.2014-
08.org.nvmeexpress:uuid:4c4c4544-0035-5910-804b-b7c04f444d33
    Transport Protocol: nvme-tcp
    Initiator Transport Address: 192.168.39.20
    Transport Service Identifier: 8009
    Host Identifier: 4c4c454400355910804bb7c04f444d33
    Admin Queue Depth: 32
    Header Digest Enabled: false
    Data Digest Enabled: false
    Keep-Alive Timeout (msec): 30000
```

Étape 9 : Configurer une authentification sécurisée en bande

L'authentification sécurisée en bande est prise en charge via NVMe/TCP entre un hôte SUSE Linux Enterprise Server 16 et un contrôleur ONTAP.

Chaque hôte ou contrôleur doit être associé à un DH-HMAC-CHAP clé pour configurer une authentification sécurisée. Une clé DH-HMAC-CHAP est une combinaison du NQN de l'hôte ou du contrôleur NVMe et d'un secret d'authentification configuré par l'administrateur. Pour authentifier son homologue, un hôte ou un contrôleur NVMe doit reconnaître la clé associée à cet homologue.

Étapes

Configurez une authentification intrabande sécurisée à l'aide de l'interface de ligne de commande ou d'un fichier de configuration JSON. Si vous devez spécifier différentes clés dhchap pour différents sous-systèmes, vous devez utiliser un fichier JSON de configuration.

CLI

Configurez l'authentification intrabande sécurisée à l'aide de l'interface de ligne de commande.

1. Obtenir le NQN hôte :

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

2. Générer la clé dhchap pour l'hôte.

Le résultat suivant décrit les gen-dhchap-key paramètres de commande :

```
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
```

Dans l'exemple suivant, une clé dhchap aléatoire avec HMAC définie sur 3 (SHA-512) est générée.

```
nvme gen-dhchap-key -m 3 -n nqn.2014-
08.org.nvmeexpress:uuid:4c4c4544-0035-5910-804b-b7c04f444d33
DHHC-
1:03:ohdxI1yIS8gBLwIOubcwI57rXcozYuRgBsoWaBvxEvpDlQHn/7dQ4JjFGwmhgwd
JWmVoripbWbMJy5eMAbCahN4hhYU=:
```

3. Sur le contrôleur ONTAP, ajoutez l'hôte et spécifiez les deux clés 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 hôte prend en charge deux types de méthodes d'authentification, unidirectionnelles et bidirectionnelles. Sur l'hôte, connectez-vous au contrôleur ONTAP et spécifiez des clés dhchap en fonction de la méthode d'authentification choisie :

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

5. Valider le nvme connect authentication en vérifiant les clés dhchap de l'hôte et du contrôleur :

a. Vérifiez les clés dhchap hôte :

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

Affiche un exemple de sortie pour une configuration unidirectionnelle

```
# cat /sys/class/nvme-subsystem/nvme-  
subsys1/nvme*/dhchap_secret  
DHHC-1:01:wkwAKk8r9Ip7qECKt7V5aIo/7Y1CH7DWkUfLfMxmseg39DFb:  
DHHC-1:01:wkwAKk8r9Ip7qECKt7V5aIo/7Y1CH7DWkUfLfMxmseg39DFb:  
DHHC-1:01:wkwAKk8r9Ip7qECKt7V5aIo/7Y1CH7DWkUfLfMxmseg39DFb:  
DHHC-1:01:wkwAKk8r9Ip7qECKt7V5aIo/7Y1CH7DWkUfLfMxmseg39DFb:
```

b. Vérifiez les clés dhchap du contrôleur :

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

Affiche un exemple de sortie pour une configuration bidirectionnelle

```
# cat /sys/class/nvme-subsystem/nvme-  
subsys6/nvme*/dhchap_ctrl_secret  
DHHC-  
1:03:ohdxI1yIS8gBLwIOubcw157rXcozYuRgBsoWaBvxEvpDlQHn/7dQ4JjFG  
wmhgwdJWmVoripbWbMJy5eMAbCahN4hhYU=:  
DHHC-  
1:03:ohdxI1yIS8gBLwIOubcw157rXcozYuRgBsoWaBvxEvpDlQHn/7dQ4JjFG  
wmhgwdJWmVoripbWbMJy5eMAbCahN4hhYU=:  
DHHC-  
1:03:ohdxI1yIS8gBLwIOubcw157rXcozYuRgBsoWaBvxEvpDlQHn/7dQ4JjFG  
wmhgwdJWmVoripbWbMJy5eMAbCahN4hhYU=:  
DHHC-  
1:03:ohdxI1yIS8gBLwIOubcw157rXcozYuRgBsoWaBvxEvpDlQHn/7dQ4JjFG  
wmhgwdJWmVoripbWbMJy5eMAbCahN4hhYU=:
```

JSON

Lorsque plusieurs sous-systèmes NVMe sont disponibles dans la configuration du contrôleur ONTAP, vous pouvez utiliser le `/etc/nvme/config.json` fichier avec la `nvme connect-all` commande.

Utilisez le `-o` option pour générer le fichier JSON. Reportez-vous aux pages de manuel NVMe connect-all pour plus d'options de syntaxe.

1. Configurez le fichier JSON :

Affiche un exemple de résultat

```
# cat /etc/nvme/config.json
[
  {
    "hostnqn": "nqn.2014-08.org.nvmeexpress:uuid:4c4c4544-0035-5910-804b-b7c04f444d33",
    "hostid": "4c4c4544-0035-5910-804b-b7c04f444d33",
    "dhchap_key": "DHHC-1:01:wkwAKk8r9Ip7qECKt7V5aIo/7Y1CH7DWkUfLfMxmseg39DFb:",
    "subsystems": [
      {
        "nqn": "nqn.1992-08.com.netapp:sn.9927e165694211f0b4f4d039eab31e9d:subsystem.inband_bidirectional",
        "ports": [
          {
            "transport": "tcp",
            "traddr": "192.168.38.10",
            "host_traddr": "192.168.38.20",
            "trsvcid": "4420",
            "dhchap_ctrl_key": "DHHC-1:03:ohdxI1yIS8gBLwIOubcw157rXcozYuRgBsoWaBvxEvpD1QHn/7dQ4JjFGwmhgwdJWmVoripbWbMJy5eMAbCahN4hhYU="
          },
          {
            "transport": "tcp",
            "traddr": "192.168.38.11",
            "host_traddr": "192.168.38.20",
            "trsvcid": "4420",
            "dhchap_ctrl_key": "DHHC-1:03:ohdxI1yIS8gBLwIOubcw157rXcozYuRgBsoWaBvxEvpD1QHn/7dQ4JjFGwmhgwdJWmVoripbWbMJy5eMAbCahN4hhYU="
          },
          {
            "transport": "tcp",
            "traddr": "192.168.39.11",
            "host_traddr": "192.168.39.20",
            "trsvcid": "4420",
            "dhchap_ctrl_key": "DHHC-1:03:ohdxI1yIS8gBLwIOubcw157rXcozYuRgBsoWaBvxEvpD1QHn/7dQ4JjFGwmhgwdJWmVoripbWbMJy5eMAbCahN4hhYU="
          },
          {
            "transport": "tcp",
            "traddr": "192.168.39.10",
            "host_traddr": "192.168.39.20",
            "trsvcid": "4420",
            "dhchap_ctrl_key": "DHHC-1:03:ohdxI1yIS8gBLwIOubcw157rXcozYuRgBsoWaBvxEvpD1QHn/7dQ4JjFGwmhgwdJWmVoripbWbMJy5eMAbCahN4hhYU="
          }
        ]
      }
    ]
  }
]
```

```
        "host_traddr": "192.168.39.20",
        "trsvcid": "4420",
        "dhchap_ctrl_key": "DHHC-
1:03:ohdxI1yIS8gBLwIOubcw157rXcozYuRgBsoWaBvxEvpD1QHn/7dQ4JjFGwm
hgwdJWmVoripbWbMJy5eMAbCahN4hhYU=:"
    }
]
}
]
}
]
```



Dans l'exemple suivant, `dhchap_key` correspond à `dhchap_secret` et `dhchap_ctrl_key` correspond à `dhchap_ctrl_secret`.

2. Connectez-vous au contrôleur ONTAP à l'aide du fichier JSON de configuration :

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

Affiche un exemple de résultat

```
traddr=192.168.38.10is already connected
traddr=192.168.39.10 is already connected
traddr=192.168.38.11 is already connected
traddr=192.168.39.11 is already connected
traddr=192.168.38.10is already connected
traddr=192.168.39.10 is already connected
traddr=192.168.38.11 is already connected
traddr=192.168.39.11 is already connected
traddr=192.168.38.10is already connected
traddr=192.168.39.10 is already connected
traddr=192.168.38.11 is already connected
traddr=192.168.39.11 is already connected
```

3. Vérifiez que les secrets dhchap ont été activés pour les contrôleurs respectifs de chaque sous-système :

- a. Vérifiez les clés dhchap hôte :

```
cat /sys/class/nvme-subsystem/nvme-subsy0/nvme0/dhchap_secret
```

L'exemple suivant montre une clé dhchap :

```
DHHC-1:01:wkwAKk8r9Ip7qECKt7V5aIo/7Y1CH7DWkUfLfMxmseg39DFb:
```

- b. Vérifiez les clés dhchap du contrôleur :

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

Vous devriez voir une sortie similaire à l'exemple suivant :

```
DHHC-
1:03:ohdxI1yIS8gBLwIOubcw157rXcozYuRgBsoWaBvxEvpDlQHn/7dQ4JjFGwmhgwd
JWmVoripbWbMJy5eMAbCahN4hhYU=:
```

Étape 10 : Configurer la sécurité de la couche de transport

Le protocole TLS (Transport Layer Security) assure un chiffrement de bout en bout sécurisé pour les connexions NVMe entre les hôtes NVMe-oF et une baie ONTAP. Vous pouvez configurer TLS 1.3 à l'aide de l'interface de ligne de commande (CLI) et d'une clé pré-partagée (PSK) configurée.

 Effectuez les étapes suivantes sur l'hôte SUSE Linux Enterprise Server, sauf indication contraire précisant que vous devez effectuer une étape sur le contrôleur ONTAP .

Étapes

1. Vérifiez que vous disposez des éléments suivants `ktls-utils`, `openssl`, et `libopenssl` paquets installés sur l'hôte :

- a. Vérifiez le `ktls-utils` :

```
rpm -qa | grep ktls
```

Vous devriez voir la sortie suivante affichée :

```
ktls-utils-0.10+33.g311d943-160000.2.2.x86_64
```

- a. Vérifiez les packages SSL :

```
rpm -qa | grep ssl
```

Affiche un exemple de résultat

```
libopenssl13-3.5.0-160000.3.2.x86_64
openssl-3.5.0-160000.2.2.noarch
openssl-3-3.5.0-160000.3.2.x86_64
libopenssl13-x86-64-v3-3.5.0-160000.3.2.x86_64
```

2. Vérifiez que vous disposez de la configuration correcte pour /etc/tlshd.conf:

```
cat /etc/tlshd.conf
```

Affiche un exemple de résultat

```
[debug]
loglevel=0
tls=0
nl=0

[authenticate]
#keyrings= <keyring>;<keyring>;<keyring>

[authenticate.client]
#x509.truststore= <pathname>
#x509.certificate= <pathname>
#x509.private_key= <pathname>

[authenticate.server]
#x509.truststore= <pathname>
#x509.certificate= <pathname>
#x509.private_key= <pathname>
```

3. Activer tlshd pour démarrer au démarrage du système :

```
systemctl enable tlshd
```

4. Vérifiez que le tlshd démon est en cours d'exécution :

```
systemctl status tlshd
```

Affiche un exemple de résultat

```
tlshd.service - Handshake service for kernel TLS consumers
   Loaded: loaded (/usr/lib/systemd/system/tlshd.service; enabled;
   preset: disabled)
     Active: active (running) since Wed 2024-08-21 15:46:53 IST; 4h
      57min ago
       Docs: man:tlshd(8)
     Main PID: 961 (tlshd)
        Tasks: 1
         CPU: 46ms
        CGroup: /system.slice/tlshd.service
                  └─961 /usr/sbin/tlshd
Aug 21 15:46:54 RX2530-M4-17-153 tlshd[961]: Built from ktls-utils
0.11-dev on Mar 21 2024 12:00:00
```

5. Générez le TLS PSK en utilisant nvme gen-tls-key :

a. Vérifiez l'hôte :

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

Vous devriez voir le résultat suivant :

```
nqn.2014-08.org.nvmeexpress:uuid:4c4c4544-0035-5910-804b-b7c04f444d33
```

b. Vérifiez la clé :

```
nvme gen-tls-key --hmac=1 --identity=1 --subsysnqn= nqn.1992-
08.com.netapp:sn.9927e165694211f0b4f4d039eab31e9d:subsystem.nvme1
```

Vous devriez voir le résultat suivant :

```
NVMeTLSkey-1:01:C50EsaGtuOp8n5fGE9EuWjbBCtshmfoHx4XTqTJUmydf0gIj:
```

6. Sur le contrôleur ONTAP, ajoutez le protocole TLS PSK au sous-système ONTAP :

Affiche un exemple de résultat

```
nvme subsystem host add -vserver vs_iscsi_tcp -subsystem nvme1 -host -nqn nqn.2014-08.org.nvmexpress:uuid:4c4c4544-0035-5910-804b-b2c04f444d33 -tls-configured-psk NVMeTLSkey-1:01:C50EsaGtuOp8n5fGE9EuWjbBCtshmfoHx4XTqTJUmydf0gIj:
```

7. Insérez le TLS PSK dans le porte-clés du noyau hôte :

```
nvme check-tls-key --identity=1 --subsysnqn=nqn.1992-08.com.netapp:sn.9927e165694211f0b4f4d039eab31e9d:subsystem.nvme1 --keydata=NVMeTLSkey-1:01:C50EsaGtuOp8n5fGE9EuWjbBCtshmfoHx4XTqTJUmydf0gIj: --insert
```

Vous devriez voir la clé TLS suivante :

```
Inserted TLS key 069f56bb
```



Le PSK montre comme NVMe1R01 parce qu'il utilise identity v1 à partir de l'algorithme de négociation TLS. Identity v1 est la seule version prise en charge par ONTAP.

8. Vérifiez que TLS PSK est correctement inséré :

```
cat /proc/keys | grep NVMe
```

Affiche un exemple de résultat

```
069f56bb I-Q-- 5 perm 3b010000 0 0 psk NVMe1R01 nqn.2014-08.org.nvmexpress:uuid:4c4c4544-0035-5910-804b-b2c04f444d33 nqn.1992-08.com.netapp:sn.9927e165694211f0b4f4d039eab31e9d:subsystem.nvme1 oYVLe1miOwnvDjXKBmrnIgGVpFIBDJtc4hmQXE/36Sw=: 32
```

9. Connectez-vous au sous-système ONTAP à l'aide du protocole TLS PSK inséré :

- Vérifiez le TLS PSK :

```
nvme connect -t tcp -w 192.168.38.20 -a 192.168.38.10 -n nqn.1992-08.com.netapp:sn.9927e165694211f0b4f4d039eab31e9d:subsystem.nvme1 --tls_key=0x069f56bb -tls
```

Vous devriez voir le résultat suivant :

```
connecting to device: nvme0
```

a. Vérifiez la liste-sous-système :

```
nvme list-subsy
```

Affiche un exemple de résultat

```
nvme-subsy0 - NQN=nqn.1992-08.com.netapp:sn.9927e165694211f0b4f4d039eab31e9d:subsystem.nvme1 hostnqn=nqn.2014-08.org.nvmeexpress:uuid:4c4c4544-0035-5910-804b-b2c04f444d33
**
+- nvme0 tcp
traddr=192.168.38.10,trsvcid=4420,host_traddr=192.168.38.20,src_a
ddr=192.168.38.20 live
```

10. Ajoutez la cible et vérifiez la connexion TLS au sous-système ONTAP spécifié :

```
nvme subsystem controller show -vserver vs_tcp_sles16 -subsystem nvme1 -instance
```

Affiche un exemple de résultat

```
(vserver nvme subsystem controller show)
    Vserver Name: vs_tcp_sles16
        Subsystem: nvme1
            Controller ID: 0040h
                Logical Interface: lif1
                    Node: A400-12-171
                    Host NQN: nqn.2014-
08.org.nvmeexpress:uuid:4c4c4544-0035-5910-804b-b2c04f444d33
                    Transport Protocol: nvme-tcp
                    Initiator Transport Address: 192.168.38.20
                    Host Identifier:
4c4c454400355910804bb2c04f444d33
                    Number of I/O Queues: 2
                    I/O Queue Depths: 128, 128
                    Admin Queue Depth: 32
                    Max I/O Size in Bytes: 1048576
                    Keep-Alive Timeout (msec): 5000
                    Subsystem UUID: 62203cf8-826a-11f0-966e-
d039eab31e9d
                    Header Digest Enabled: false
                    Data Digest Enabled: false
                    Authentication Hash Function: sha-256
                    Authentication Diffie-Hellman Group: 3072-bit
                    Authentication Mode: unidirectional
                    Transport Service Identifier: 4420
                    TLS Key Type: configured
                    TLS PSK Identity: NVMe1R01 nqn.2014-
08.org.nvmeexpress:uuid:4c4c4544-0035-5910-804b-b2c04f444d33
nqn.1992-
08.com.netapp:sn.9927e165694211f0b4f4d039eab31e9d:subsystem.nvme1
oYVLelmiOwnvDjXKBmrnIgGVpFIBDJtc4hmQXE/36Sw=
                    TLS Cipher: TLS-AES-128-GCM-SHA256
```

Étape 11 : passez en revue les problèmes connus

Il n'y a pas de problème connu.

Configurer SUSE Linux Enterprise Server 15 SPx pour NVMe-oF avec stockage ONTAP

L'hôte SUSE Linux Enterprise Server 15 SPx prend en charge les protocoles NVMe over Fibre Channel (NVMe/FC) et NVMe over TCP (NVMe/TCP) avec Asymmetric

Namespace Access (ANA). ANA offre une fonctionnalité de chemin multiple équivalente à l'accès asymétrique aux unités logiques (ALUA) dans les environnements iSCSI et FCP.

Découvrez comment configurer les hôtes NVMe over Fabrics (NVMe-oF) pour SUSE Linux Enterprise Server 15 SPx. Pour plus d'assistance et d'informations sur les fonctionnalités, consultez "["Prise en charge et fonctionnalités ONTAP"](#).

NVMe-oF avec SUSE Linux Enterprise Server 15 SPx présente les limitations connues suivantes :

- Le `nvme disconnect-all` Cette commande déconnecte les systèmes de fichiers racine et de données et peut entraîner une instabilité du système. Ne pas utiliser cette commande sur des systèmes démarrant à partir d'un SAN via des espaces de noms NVMe-TCP ou NVMe-FC.
- La prise en charge de l'utilitaire hôte `sanlun` de NetApp n'est pas disponible pour NVMe-oF. Vous pouvez en revanche utiliser le plug-in NetApp inclus dans le logiciel natif. `nvme-cli` package pour tous les transports NVMe-oF.
- Pour SUSE Linux Enterprise Server 15 SP6 et versions antérieures, le démarrage SAN utilisant le protocole NVMe-oF n'est pas pris en charge.

Étape 1 : activez éventuellement le démarrage SAN

Vous pouvez configurer votre hôte pour utiliser le démarrage SAN afin de simplifier le déploiement et d'améliorer l'évolutivité. Utilisez le "["Matrice d'interopérabilité"](#)" pour vérifier que votre système d'exploitation Linux, votre adaptateur de bus hôte (HBA), votre micrologiciel HBA, votre BIOS de démarrage HBA et votre version ONTAP prennent en charge le démarrage SAN.

Étapes

1. "["Créez un espace de noms NVMe et mappez-le à l'hôte"](#) .
2. Activez le démarrage SAN dans le BIOS du serveur pour les ports auxquels l'espace de noms de démarrage SAN est mappé.

Pour plus d'informations sur l'activation du BIOS HBA, reportez-vous à la documentation spécifique au fournisseur.

3. Redémarrez l'hôte et vérifiez que le système d'exploitation est opérationnel.

Étape 2 : Installez SUSE Linux Enterprise Server et le logiciel NVMe, puis vérifiez votre configuration.

Pour configurer votre hôte pour NVMe-oF, vous devez installer les packages logiciels hôte et NVMe, activer le multipathing et vérifier la configuration NQN de votre hôte.

Étapes

1. Installez SUSE Linux Enterprise Server 15 SPx sur le serveur. Une fois l'installation terminée, vérifiez que vous utilisez bien le noyau SUSE Linux Enterprise Server 15 SPx spécifié :

```
uname -r
```

Exemple de version du noyau Rocky Linux :

```
6.4.0-150700.53.3-default
```

2. Installer le nvme-cli groupe :

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

L'exemple suivant montre un nvme-cli version du paquet :

```
nvme-cli-2.11+22.gd31b1a01-150700.3.3.2.x86_64
```

3. Installer le libnvme groupe :

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

L'exemple suivant montre un libnvme version du paquet :

```
libnvme1-1.11+4.ge68a91ae-150700.4.3.2.x86_64
```

4. Sur l'hôte, vérifiez la chaîne hostnqn à /etc/nvme/hostnqn :

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

L'exemple suivant montre un hostnqn version:

```
nqn.2014-08.org.nvmeexpress:uuid:f6517cae-3133-11e8-bbff-7ed30aef123f
```

5. Sur le système ONTAP , vérifiez que le hostnqn La chaîne correspond à hostnqn chaîne de caractères pour le sous-système correspondant sur la matrice ONTAP :

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

Montrer l'exemple

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



Si le hostnqn les chaînes ne correspondent pas, utilisez la commande `vserver modify` pour mettre à jour le hostnqn Chaîne sur le sous-système de matrice ONTAP correspondant à hostnqn chaîne de /etc/nvme/hostnqn sur l'hôte.

Étape 3 : Configurer NVMe/FC et NVMe/TCP

Configurez NVMe/FC avec des adaptateurs Broadcom/Emulex ou Marvell/QLogic, ou configurez NVMe/TCP à l'aide d'opérations de découverte et de connexion manuelles.

NVMe/FC - Broadcom/Emulex

Configuration de NVMe/FC pour une carte FC Broadcom/Emulex

Étapes

1. Vérifiez que vous utilisez le modèle d'adaptateur pris en charge :

- a. Afficher les noms des modèles :

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

Vous devriez voir le résultat suivant :

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

- b. Afficher les descriptions des modèles :

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

Vous devriez voir le résultat suivant :

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

2. Vérifiez que vous utilisez la carte Broadcom recommandée lpfc micrologiciel et pilote de boîte de réception :

- a. Afficher la version du firmware :

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

L'exemple suivant montre les versions du firmware :

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

- b. Afficher la version du pilote de la boîte de réception :

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

L'exemple suivant montre une version de pilote :

```
0:14.4.0.8
```

Pour obtenir la liste actuelle des versions de pilotes et de micrologiciels de carte prises en charge, consultez le "[Matrice d'interopérabilité](#)".

3. Vérifiez que la sortie attendue de `lpfc_enable_fc4_type` est définie sur 3:

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

4. Vérifiez que vous pouvez afficher vos ports initiateurs :

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

Vous devriez voir une sortie similaire à :

```
0x10000090fae0ec88  
0x10000090fae0ec89
```

5. Vérifiez que vos ports initiateurs sont en ligne :

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

Vous devriez voir le résultat suivant :

```
Online  
Online
```

6. Vérifiez que les ports initiateurs NVMe/FC sont activés et que les ports cibles sont visibles :

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

Affiche un exemple de résultat

```
NVME Initiator Enabled
XRI Dist lpfco Total 6144 IO 5894 ELS 250
NVME LPORT lpfco WWPN x10000090fae0ec88 WWNN x20000090fae0ec88
DID x0a1300 ONLINE
NVME RPORT WWPN x23b1d039ea359e4a WWNN x23aed039ea359e4a
DID x0a1c01 TARGET DISCSRVC ONLINE
NVME RPORT WWPN x22bb0d039ea359e4a WWNN x22b8d039ea359e4a
DID x0a1c0b TARGET DISCSRVC ONLINE
NVME RPORT WWPN x2362d039ea359e4a WWNN x234ed039ea359e4a
DID x0a1c10 TARGET DISCSRVC ONLINE
NVME RPORT WWPN x23afd039ea359e4a WWNN x23aed039ea359e4a
DID x0a1a02 TARGET DISCSRVC ONLINE
NVME RPORT WWPN x22b9d039ea359e4a WWNN x22b8d039ea359e4a
DID x0a1a0b TARGET DISCSRVC ONLINE
NVME RPORT WWPN x2360d039ea359e4a WWNN x234ed039ea359e4a
DID x0a1a11 TARGET DISCSRVC ONLINE
```

```
NVME Statistics
LS: Xmt 0000004ea0 Cmpl 0000004ea0 Abort 00000000
LS XMIT: Err 00000000 CMPL: xb 00000000 Err 00000000
Total FCP Cmpl 0000000000102c35 Issue 0000000000102c2d OutIO
ffffffffffff8
    abort 00000175 noxri 00000000 nondlp 0000021d qdepth
00000000 wqerr 00000007 err 00000000
FCP CMPL: xb 00000175 Err 0000058b
```

```
NVME Initiator Enabled
XRI Dist lpfcl Total 6144 IO 5894 ELS 250
NVME LPORT lpfcl WWPN x10000090fae0ec89 WWNN x20000090fae0ec89
DID x0a1200 ONLINE
NVME RPORT WWPN x23b2d039ea359e4a WWNN x23aed039ea359e4a
DID x0a1d01 TARGET DISCSRVC ONLINE
NVME RPORT WWPN x22bcd039ea359e4a WWNN x22b8d039ea359e4a
DID x0a1d0b TARGET DISCSRVC ONLINE
NVME RPORT WWPN x2363d039ea359e4a WWNN x234ed039ea359e4a
DID x0a1d10 TARGET DISCSRVC ONLINE
NVME RPORT WWPN x23b0d039ea359e4a WWNN x23aed039ea359e4a
DID x0a1b02 TARGET DISCSRVC ONLINE
NVME RPORT WWPN x22bad039ea359e4a WWNN x22b8d039ea359e4a
DID x0a1b0b TARGET DISCSRVC ONLINE
NVME RPORT WWPN x2361d039ea359e4a WWNN x234ed039ea359e4a
DID x0a1b11 TARGET DISCSRVC ONLINE
```

```
NVME Statistics
```

```
LS: Xmt 0000004e31 Cmpl 0000004e31 Abort 00000000
LS XMIT: Err 00000000 CMPL: xb 00000000 Err 00000000
Total FCP Cmpl 00000000001017f2 Issue 00000000001017ef OutIO
fffffffffffffd
    abort 0000018a noxri 00000000 nondlp 0000012e qdepth
00000000 wqerr 00000004 err 00000000
FCP CMPL: xb 0000018a Err 000005ca
```

NVMe/FC - Marvell/QLogic

Configuration du NVMe/FC pour un adaptateur Marvell/QLogic

Étapes

1. Vérifiez que vous exécutez les versions du pilote de carte et du micrologiciel prises en charge :

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

L'exemple suivant montre les versions du pilote et du micrologiciel :

```
QLE2742 FW:v9.14.00 DVR:v10.02.09.400-k-debug
QLE2742 FW:v9.14.00 DVR:v10.02.09.400-k-debug
```

2. Vérifiez-le `ql2xnvmeenable` est défini. L'adaptateur Marvell peut ainsi fonctionner en tant qu'initiateur NVMe/FC :

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

La sortie attendue est 1.

NVMe/TCP

Le protocole NVMe/TCP ne prend pas en charge l'opération de connexion automatique. Au lieu de cela, vous pouvez découvrir les sous-systèmes et espaces de noms NVMe/TCP en exécutant l'opération NVMe/TCP. connect ou connect-all opérations manuellement.

Étapes

1. Vérifiez que le port initiateur peut récupérer les données de la page de journal de découverte sur les LIF NVMe/TCP prises en charge :

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

Affiche un exemple de résultat

```
nvme discover -t tcp -w 192.168.111.80 -a 192.168.111.70

Discovery Log Number of Records 8, Generation counter 42
=====Discovery Log Entry 0=====
trtype: tcp
adrifam: ipv4
subtype: current discovery subsystem
treq: not specified
portid: 4
trsvcid: 8009
subnqn: nqn.1992-
08.com.netapp:sn.f8e2af201b7211f0ac2bd039eab67a95:discovery
traddr: 192.168.211.71
eflags: explicit discovery connections, duplicate discovery
information
sectype: none
=====Discovery Log Entry 1=====
trtype: tcp
adrifam: ipv4
subtype: current discovery subsystem
treq: not specified
portid: 3
trsvcid: 8009
subnqn: nqn.1992-
08.com.netapp:sn.f8e2af201b7211f0ac2bd039eab67a95:discovery
traddr: 192.168.111.71
eflags: explicit discovery connections, duplicate discovery
information
sectype: none
=====Discovery Log Entry 2=====
trtype: tcp
adrifam: ipv4
subtype: current discovery subsystem
treq: not specified
portid: 2
trsvcid: 8009
subnqn: nqn.1992-
08.com.netapp:sn.f8e2af201b7211f0ac2bd039eab67a95:discovery
traddr: 192.168.211.70
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.f8e2af201b7211f0ac2bd039eab67a95:discovery
traddr: 192.168.111.70
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.f8e2af201b7211f0ac2bd039eab67a95:subsystem.samp
le_tcp_sub
traddr: 192.168.211.71
eflags: none
sectype: none
=====Discovery Log Entry 5=====
trtype: tcp
adrfam: ipv4
subtype: nvme subsystem
treq: not specified
portid: 3
trsvcid: 4420
subnqn: nqn.1992-
08.com.netapp:sn.f8e2af201b7211f0ac2bd039eab67a95:subsystem.samp
le_tcp_sub
traddr: 192.168.111.71
eflags: none
sectype: none
=====Discovery Log Entry 6=====
trtype: tcp
adrfam: ipv4
subtype: nvme subsystem
treq: not specified
portid: 2
trsvcid: 4420
subnqn: nqn.1992-
08.com.netapp:sn.f8e2af201b7211f0ac2bd039eab67a95:subsystem.samp
```

```
le_tcp_sub
traddr: 192.168.211.70
eflags: none
sectype: none
=====Discovery Log Entry 7=====
trtype: tcp
adrifam: ipv4
subtype: nvme subsystem
treq: not specified
portid: 1
trsvcid: 4420
subnqn: nqn.1992-
08.com.netapp:sn.f8e2af201b7211f0ac2bd039eab67a95:subsystem.samp
le_tcp_sub
traddr: 192.168.111.70
eflags: none
sectype: none
localhost:~ #
```

2. Vérifier que toutes les autres combinaisons de LIF NVMe/TCP initiator-target peuvent récupérer les données de la page du journal de découverte :

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

Montrer l'exemple

```
nvme discover -t tcp -w 192.168.111.80 -a 192.168.111.66
nvme discover -t tcp -w 192.168.111.80 -a 192.168.111.67
nvme discover -t tcp -w 192.168.211.80 -a 192.168.211.66
nvme discover -t tcp -w 192.168.211.80 -a 192.168.211.67
```

3. Exécutez la commande `nvme connect-all` sur toutes les LIF cible-initiateur NVMe/TCP prises en charge sur l'ensemble des nœuds :

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

Montrer l'exemple

```
nvme connect-all -t tcp -w 192.168.111.80 -a  
192.168.111.66  
nvme connect-all -t tcp -w 192.168.111.80 -a  
192.168.111.67  
nvme connect-all -t tcp -w 192.168.211.80 -a  
192.168.211.66  
nvme connect-all -t tcp -w 192.168.211.80 -a  
192.168.211.67
```

À partir de SUSE Linux Enterprise Server 15 SP6, le paramètre NVMe/TCP `ctrl_loss_tmo timeout` est automatiquement réglé sur « désactivé ». Par conséquent:

- Il n'y a pas de limite au nombre de tentatives (nouvelle tentative indéfinie).
- Vous n'avez pas besoin de configurer manuellement un élément spécifique `ctrl_loss_tmo timeout` durée lors de l'utilisation du `nvme connect` ou `nvme connect-all` commandes (option `-l`).
- Les contrôleurs NVMe/TCP ne subissent pas de dépassement de délai en cas de défaillance d'un chemin et restent connectés indéfiniment.

Étape 4 : Vous pouvez éventuellement modifier la politique d'E/S dans les règles udev.

À partir de SUSE Linux Enterprise Server 15 SP6, la stratégie d'E/S par défaut pour NVMe-oF est définie sur `round-robin`. Si vous souhaitez modifier la politique d'E/S en `queue-depth`, modifiez le fichier de règles udev comme suit :

Étapes

1. Ouvrez le fichier de règles udev dans un éditeur de texte avec des privilèges root :

```
/usr/lib/udev/rules.d/71-nvme.rules
```

Vous devriez voir le résultat suivant :

```
vi /usr/lib/udev/rules.d/71-nvme.rules
```

2. Recherchez la ligne qui définit la politique d'E/S pour le contrôleur NetApp ONTAP, comme indiqué dans l'exemple de règle suivant :

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

3. Modifier la règle afin que round-robin devient queue-depth :

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

4. Rechargez les règles udev et appliquez les modifications :

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

5. Vérifiez la politique d'E/S actuelle de votre sous-système. Remplacez <subsystem>, par exemple, nvme-subsys0 .

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

Vous devriez voir le résultat suivant :

queue-depth.

 La nouvelle politique d'E/S s'applique automatiquement aux périphériques NetApp ONTAP Controller correspondants. Vous n'avez pas besoin de redémarrer.

Étape 5 : Vous pouvez activer l'E/S à 1 Mo pour NVMe/FC (optionnel).

ONTAP signale une taille de transfert de données maximale (MDTS) de 8 dans les données du contrôleur d'identification. Cela signifie que la taille maximale de la demande d'E/S peut atteindre 1 Mo. Pour émettre des requêtes d'E/S d'une taille de 1 Mo pour un hôte Broadcom NVMe/FC, vous devez augmenter la `1pfc` valeur de la `1pfc_sg_seg_cnt` paramètre à 256 à partir de la valeur par défaut de 64.

 Ces étapes ne s'appliquent pas aux hôtes NVMe/FC Qlogic.

Étapes

1. Réglez le `1pfc_sg_seg_cnt` paramètre sur 256 :

```
cat /etc/modprobe.d/1pfc.conf
```

Vous devriez voir une sortie similaire à l'exemple suivant :

```
options lpfc lpfc_sg_seg_cnt=256
```

2. Exécutez `dracut -f` la commande et redémarrez l'hôte.
3. Vérifier que la valeur de `lpfc_sg_seg_cnt` est 256 :

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

Étape 6 : Vérifier les services de démarrage NVMe

Le `nvmefc-boot-connections.service` et `nvmf-autoconnect.service` services de démarrage inclus dans NVMe/FC `nvme-cl` les packages sont automatiquement activés au démarrage du système.

Une fois le démarrage terminé, vérifiez que le `nvmefc-boot-connections.service` et `nvmf-autoconnect.service` les services de démarrage sont activés.

Étapes

1. Vérifiez que `nvmf-autoconnect.service` est activé :

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

Affiche un exemple de résultat

```
nvmf-autoconnect.service - Connect NVMe-oF subsystems automatically
during boot
   Loaded: loaded (/usr/lib/systemd/system/nvmf-autoconnect.service;
             enabled; preset: enabled)
   Active: inactive (dead) since Fri 2025-07-04 23:56:38 IST; 4 days
             ago
     Main PID: 12208 (code=exited, status=0/SUCCESS)
       CPU: 62ms

Jul  4 23:56:26 localhost systemd[1]: Starting Connect NVMe-oF
subsystems automatically during boot...
Jul  4 23:56:38 localhost systemd[1]: nvmf-autoconnect.service:
Deactivated successfully.
Jul  4 23:56:38 localhost systemd[1]: Finished Connect NVMe-oF
subsystems automatically during boot.
```

2. Vérifiez que `nvmefc-boot-connections.service` est activé :

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

Affiche un exemple de résultat

```
nvmefc-boot-connections.service - Auto-connect to subsystems on FC-NVME devices found during boot
   Loaded: loaded (/usr/lib/systemd/system/nvmefc-boot-connections.service; enabled; preset: enabled)
   Active: inactive (dead) since Mon 2025-07-07 19:52:30 IST; 1 day 4h ago
     Main PID: 2945 (code=exited, status=0/SUCCESS)
       CPU: 14ms

Jul 07 19:52:30 HP-DL360-14-168 systemd[1]: Starting Auto-connect to subsystems on FC-NVME devices found during boot...
Jul 07 19:52:30 HP-DL360-14-168 systemd[1]: nvmefc-boot-connections.service: Deactivated successfully.
Jul 07 19:52:30 HP-DL360-14-168 systemd[1]: Finished Auto-connect to subsystems on FC-NVME devices found during boot.
```

Étape 7 : Vérifier la configuration du multipathing

Vérifiez que l'état des chemins d'accès multiples NVMe in-kernel, l'état ANA et les namespaces ONTAP sont corrects pour la configuration NVMe-of.

Étapes

1. Vérifiez que le chemin d'accès multiples NVMe intégré au noyau est activé :

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

Vous devriez voir le résultat suivant :

```
Y
```

2. Vérifiez que les paramètres NVMe-oF appropriés (tels que le modèle défini sur NetApp ONTAP Controller et la stratégie d'E/S d'équilibrage de charge définie sur queue-depth) pour les espaces de noms ONTAP respectifs se reflètent correctement sur l'hôte :

- a. Afficher les sous-systèmes :

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

Vous devriez voir le résultat suivant :

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

b. Afficher la politique :

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

Vous devriez voir le résultat suivant :

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

3. Vérifiez que les espaces de noms sont créés et correctement découverts sur l'hôte :

```
nvme list
```

Montrer l'exemple

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

Namespace	Usage	Format	FW	Rev
1	21.47 GB / 21.47 GB	4 KiB + 0 B	FFFFFFFFFF	

4. Vérifiez que l'état du contrôleur de chaque chemin est actif et que l'état ANA est correct :

NVMe/FC

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

Affiche un exemple de résultat

```
nvme-subsys114 - NQN=nqn.1992-
08.com.netapp:sn.9e30b9760a4911f08c87d039eab67a95:subsystem.sles
_161_27
    hostnqn=nqn.2014-
08.org.nvmeexpress:uuid:f6517cae-3133-11e8-bbff-7ed30aef123f
iopolicy=round-robin\
+- nvme114 fc traddr=nn-0x234ed039ea359e4a:pn-
0x2360d039ea359e4a,host_traddr=nn-0x20000090fae0ec88:pn-
0x10000090fae0ec88 live optimized
+- nvme115 fc traddr=nn-0x234ed039ea359e4a:pn-
0x2362d039ea359e4a,host_traddr=nn-0x20000090fae0ec88:pn-
0x10000090fae0ec88 live non-optimized
+- nvme116 fc traddr=nn-0x234ed039ea359e4a:pn-
0x2361d039ea359e4a,host_traddr=nn-0x20000090fae0ec89:pn-
0x10000090fae0ec89 live optimized
+- nvme117 fc traddr=nn-0x234ed039ea359e4a:pn-
0x2363d039ea359e4a,host_traddr=nn-0x20000090fae0ec89:pn-
0x10000090fae0ec89 live non-optimized
```

NVMe/TCP

```
nvme list-subsys /dev/nvme9n1
```

Affiche un exemple de résultat

```
nvme-subsy9 - NQN=nqn.1992-
08.com.netapp:sn.f8e2af201b7211f0ac2bd039eab67a95:subsystem.with
_inband_with_json hostnqn=nqn.2014-
08.org.nvmeexpress:uuid:4c4c4544-0035-5910-804b-b2c04f444d33
iopolicy=round-robin
\
+- nvme10 tcp
traddr=192.168.111.71,trsvcid=4420,src_addr=192.168.111.80 live
non-optimized
+- nvme11 tcp
traddr=192.168.211.70,trsvcid=4420,src_addr=192.168.211.80 live
optimized
+- nvme12 tcp
traddr=192.168.111.70,trsvcid=4420,src_addr=192.168.111.80 live
optimized
+- nvme9 tcp
traddr=192.168.211.71,trsvcid=4420,src_addr=192.168.211.80 live
non-optimized
```

5. Vérifier que le plug-in NetApp affiche les valeurs correctes pour chaque périphérique d'espace de noms ONTAP :

Colonne

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

Montrer l'exemple

Device	Vserver	Namespace Path
NSID	UUID	Size
/dev/nvme0n1	vs_161	
/vol/fc_nvme_vol1/fc_nvme_ns1		1
32fd92c7-0797-428e-a577-fdb3f14d0dc3		5.37GB

JSON

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

Montrer l'exemple

```
{  
    "Device": "/dev/nvme98n2",  
    "Vserver": "vs_161",  
    "Namespace_Path": "/vol/fc_nvme_vol71/fc_nvme_ns71",  
    "NSID": 2,  
    "UUID": "39d634c4-a75e-4fbd-ab00-3f9355a26e43",  
    "LBA_Size": 4096,  
    "Namespace_Size": 5368709120,  
    "UsedBytes": 430649344,  
}  
]  
}
```

Étape 8 : Créer un contrôleur de découverte persistant

Vous pouvez créer un contrôleur de découverte persistant (PDC) pour un hôte SUSE Linux Enterprise Server 15 SPx. Un contrôleur de périphérique (PDC) est nécessaire pour détecter automatiquement une opération d'ajout ou de suppression d'un sous-système NVMe et les modifications apportées aux données de la page du journal de découverte.

Étapes

1. Vérifier que les données de la page du journal de découverte sont disponibles et peuvent être récupérées via la combinaison port initiateur et LIF cible :

```
nvme discover -t <trtype> -w <host-traddr> -a <traddr>
```

Affiche un exemple de résultat

```
Discovery Log Number of Records 8, Generation counter 18
=====Discovery Log Entry 0=====
trtype: tcp
adrifam: ipv4
subtype: current discovery subsystem
treq: not specified
portid: 4
trsvcid: 8009
subnqn: nqn.1992-
08.com.netapp:sn.4f7af2bd221811f0afadd039eab0dadd:discovery
traddr: 192.168.111.66
eflags: explicit discovery connections, duplicate discovery
information
sectype: none
=====Discovery Log Entry 1=====
trtype: tcp
adrifam: ipv4
subtype: current discovery subsystem
treq: not specified
portid: 2
trsvcid: 8009
subnqn: nqn.1992-
08.com.netapp:sn.4f7af2bd221811f0afadd039eab0dadd:discovery
traddr: 192.168.211.66
eflags: explicit discovery connections, duplicate discovery
information
sectype: none
=====Discovery Log Entry 2=====
trtype: tcp
adrifam: ipv4
subtype: current discovery subsystem
treq: not specified
portid: 3
trsvcid: 8009
subnqn: nqn.1992-
08.com.netapp:sn.4f7af2bd221811f0afadd039eab0dadd:discovery
traddr: 192.168.111.67
eflags: explicit discovery connections, duplicate discovery
information
sectype: none
=====Discovery Log Entry 3=====
trtype: tcp
adrifam: ipv4
subtype: current discovery subsystem
```

```
treq:    not specified
portid:  1
trsvcid: 8009
subnqn:  nqn.1992-
08.com.netapp:sn.4f7af2bd221811f0afadd039eab0dadd:discovery
traddr:  192.168.211.67
eflags: explicit discovery connections, duplicate discovery
information
sectype: none
=====Discovery Log Entry 4=====
trtype:  tcp
adrifam:  ipv4
subtype: nvme subsystem
treq:    not specified
portid:  4
trsvcid: 4420
subnqn:  nqn.1992-
08.com.netapp:sn.4f7af2bd221811f0afadd039eab0dadd:subsystem.pdc
traddr:  192.168.111.66
eflags:  none
sectype: none
=====Discovery Log Entry 5=====
trtype:  tcp
adrifam:  ipv4
subtype: nvme subsystem
treq:    not specified
portid:  2
trsvcid: 4420
subnqn:  nqn.1992-
08.com.netapp:sn.4f7af2bd221811f0afadd039eab0dadd:subsystem.pdc
traddr:  192.168.211.66
eflags:  none
sectype: none
=====Discovery Log Entry 6=====
trtype:  tcp
adrifam:  ipv4
subtype: nvme subsystem
treq:    not specified
portid:  3
trsvcid: 4420
subnqn:  nqn.1992-
08.com.netapp:sn.4f7af2bd221811f0afadd039eab0dadd:subsystem.pdc
traddr:  192.168.111.67
eflags:  none
sectype: none
=====Discovery Log Entry 7=====
```

```
trtype:    tcp
adrifam:   ipv4
subtype:   nvme subsystem
treq:      not specified
portid:    1
trsvcid:  4420
subnqn:   nqn.1992-
08.com.netapp:sn.4f7af2bd221811f0afadd039eab0dadd:subsystem.pdc
traddr:   192.168.211.67
eflags:   none
sectype:  none
```

2. Créer un PDC pour le sous-système de découverte :

```
nvme discover -t <trtype> -w <host-traddr> -a <traddr> -p
```

Vous devriez voir le résultat suivant :

```
nvme discover -t tcp -w 192.168.111.80 -a 192.168.111.66 -p
```

3. À partir du contrôleur ONTAP, vérifier que le PDC a été créé :

```
vserver nvme show-discovery-controller -instance -vserver <vserver_name>
```

Affiche un exemple de résultat

```
vserver nvme show-discovery-controller -instance -vserver vs_pdc

    Vserver Name: vs_pdc
        Controller ID: 0101h
        Discovery Subsystem NQN: nqn.1992-
08.com.netapp:sn.4f7af2bd221811f0afadd039eab0dadd:discovery
        Logical Interface: lif2
            Node: A400-12-181
            Host NQN: nqn.2014-
08.org.nvmeexpress:uuid:9796c1ec-0d34-11eb-b6b2-3a68dd3bab57
            Transport Protocol: nvme-tcp
            Initiator Transport Address: 192.168.111.80
            Transport Service Identifier: 8009
            Host Identifier: 9796c1ec0d3411ebb6b23a68dd3bab57
            Admin Queue Depth: 32
            Header Digest Enabled: false
            Data Digest Enabled: false
            Keep-Alive Timeout (msec): 30000
```

Étape 9 : Configurer une authentification sécurisée en bande

L'authentification sécurisée en bande est prise en charge via NVMe/TCP entre un hôte SUSE Linux Enterprise Server 15 SPx et un contrôleur ONTAP .

Chaque hôte ou contrôleur doit être associé à un DH-HMAC-CHAP clé pour configurer une authentification sécurisée. Une clé DH-HMAC-CHAP est une combinaison du NQN de l'hôte ou du contrôleur NVMe et d'un secret d'authentification configuré par l'administrateur. Pour authentifier son homologue, un hôte ou un contrôleur NVMe doit reconnaître la clé associée à cet homologue.

Étapes

Configurez une authentification intrabande sécurisée à l'aide de l'interface de ligne de commande ou d'un fichier de configuration JSON. Si vous devez spécifier différentes clés dhchap pour différents sous-systèmes, vous devez utiliser un fichier JSON de configuration.

CLI

Configurez l'authentification intrabande sécurisée à l'aide de l'interface de ligne de commande.

1. Obtenir le NQN hôte :

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

2. Générer la clé dhchap pour l'hôte.

Le résultat suivant décrit les gen-dhchap-key paramètres de commande :

```
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
```

Dans l'exemple suivant, une clé dhchap aléatoire avec HMAC définie sur 3 (SHA-512) est générée.

```
nvme gen-dhchap-key -m 3 -n nqn.2014-
08.org.nvmeexpress:uuid:e6dade64-216d-11ec-b7bb-7ed30a5482c3
DHHC-
1:03:1CFivw9ccz58gAcOUJrM7Vs98hd2ZHSr+iw+Amg6xZP15D2Yk+HDTZiUAg1iGgx
TYqnxukqvYedA55Bw3wtz6sJNpR4=:
```

3. Sur le contrôleur ONTAP, ajoutez l'hôte et spécifiez les deux clés 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 hôte prend en charge deux types de méthodes d'authentification, unidirectionnelles et bidirectionnelles. Sur l'hôte, connectez-vous au contrôleur ONTAP et spécifiez des clés dhchap en fonction de la méthode d'authentification choisie :

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

5. Valider le nvme connect authentication en vérifiant les clés dhchap de l'hôte et du contrôleur :

a. Vérifiez les clés dhchap hôte :

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

Affiche un exemple de sortie pour une configuration unidirectionnelle

```
cat /sys/class/nvme-subsystem/nvme-subsy1/nvme*/dhchap_secret
DHHC-1:01:iM63E6cX7G5SOKKOju8gmzM53qywsy+C/YwtzxhIt9ZRz+ky:
DHHC-1:01:iM63E6cX7G5SOKKOju8gmzM53qywsy+C/YwtzxhIt9ZRz+ky:
DHHC-1:01:iM63E6cX7G5SOKKOju8gmzM53qywsy+C/YwtzxhIt9ZRz+ky:
DHHC-1:01:iM63E6cX7G5SOKKOju8gmzM53qywsy+C/YwtzxhIt9ZRz+ky:
```

b. Vérifiez les clés dhchap du contrôleur :

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

Affiche un exemple de sortie pour une configuration bidirectionnelle

```
cat /sys/class/nvme-subsystem/nvme-subsy6/nvme*/dhchap_ctrl_secret
DHHC-
1:03:1CFivw9ccz58gAcOUJrM7Vs98hd2ZHSr+iw+Amg6xZP15D2Yk+HDTZiUA
g1iGgxTYqnxukqvYedA55Bw3wtz6sJNpR4=:
DHHC-
1:03:1CFivw9ccz58gAcOUJrM7Vs98hd2ZHSr+iw+Amg6xZP15D2Yk+HDTZiUA
g1iGgxTYqnxukqvYedA55Bw3wtz6sJNpR4=:
DHHC-
1:03:1CFivw9ccz58gAcOUJrM7Vs98hd2ZHSr+iw+Amg6xZP15D2Yk+HDTZiUA
g1iGgxTYqnxukqvYedA55Bw3wtz6sJNpR4=:
DHHC-
1:03:1CFivw9ccz58gAcOUJrM7Vs98hd2ZHSr+iw+Amg6xZP15D2Yk+HDTZiUA
g1iGgxTYqnxukqvYedA55Bw3wtz6sJNpR4=:
```

JSON

Lorsque plusieurs sous-systèmes NVMe sont disponibles dans la configuration du contrôleur ONTAP, vous pouvez utiliser le `/etc/nvme/config.json` fichier avec la `nvme connect-all` commande.

Utilisez le `-o` option pour générer le fichier JSON. Reportez-vous aux pages de manuel NVMe connect-all pour plus d'options de syntaxe.

1. Configurez le fichier JSON :

Affiche un exemple de résultat

```
cat /etc/nvme/config.json
[
{
    "hostnqn": "nqn.2014-08.org.nvmeexpress:uuid:4c4c4544-0035-5910-804b-b2c04f444d33",
    "hostid": "4c4c4544-0035-5910-804b-b2c04f444d33",
    "dhchap_key": "DHHC-1:01:i4i789R11sMuHLCY27RVI8XloC\GzjRwyhxip5hmIELsHrBq:",
    "subsystems": [
        {
            "nqn": "nqn.1992-08.com.netapp:sn.f8e2af201b7211f0ac2bd039eab67a95:subsystem.sample_tcp_sub",
            "ports": [
                {
                    "transport": "tcp",
                    "traddr": "192.168.111.70",
                    "host_traddr": "192.168.111.80",
                    "trsvcid": "4420",
                    "dhchap_ctrl_key": "DHHC-1:03:jqgYcJSKp73+XqAf2X6twr9ngBpr2n0MGWbmZIZq4PieKZCoilKGef81AvhYS0PNK7T+04YD5CRPjh+m3qjJU++yR8s="
                },
                {
                    "transport": "tcp",
                    "traddr": "192.168.111.71",
                    "host_traddr": "192.168.111.80",
                    "trsvcid": "4420",
                    "dhchap_ctrl_key": "DHHC-1:03:jqgYcJSKp73+XqAf2X6twr9ngBpr2n0MGWbmZIZq4PieKZCoilKGef81AvhYS0PNK7T+04YD5CRPjh+m3qjJU++yR8s="
                },
                {
                    "transport": "tcp",
                    "traddr": "192.168.211.70",
                    "host_traddr": "192.168.211.80",
                    "trsvcid": "4420",
                    "dhchap_ctrl_key": "DHHC-1:03:jqgYcJSKp73+XqAf2X6twr9ngBpr2n0MGWbmZIZq4PieKZCoilKGef81AvhYS0PNK7T+04YD5CRPjh+m3qjJU++yR8s="
                },
                {
                    "transport": "tcp",
                    "traddr": "192.168.211.71",
```

```
        "host_traddr": "192.168.211.80",
        "trsvcid": "4420",
        "dhchap_ctrl_key": "DHHC-
1:03:jqgYcJSKp73+XqAf2X6twr9ngBpr2n0MGWbmZIZq4PieKZCoilKGef81Avh
YS0PNK7T+04YD5CRPjh+m3qjJU++yR8s=:"
    }
]
}
]
```



Dans l'exemple suivant, `dhchap_key` correspond à `dhchap_secret` et `dhchap_ctrl_key` correspond à `dhchap_ctrl_secret`.

2. Connectez-vous au contrôleur ONTAP à l'aide du fichier JSON de configuration :

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

Affiche un exemple de résultat

```
traddr=192.168.211.70 is already connected
traddr=192.168.111.71 is already connected
traddr=192.168.211.71 is already connected
traddr=192.168.111.70 is already connected
traddr=192.168.211.70 is already connected
traddr=192.168.111.70 is already connected
traddr=192.168.211.71 is already connected
traddr=192.168.111.71 is already connected
traddr=192.168.211.70 is already connected
traddr=192.168.111.71 is already connected
traddr=192.168.211.71 is already connected
traddr=192.168.111.70 is already connected
```

3. Vérifiez que les secrets dhchap ont été activés pour les contrôleurs respectifs de chaque sous-système :

- Vérifiez les clés dhchap hôte :

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

L'exemple suivant montre une clé dhchap :

```
DHHC-1:01:i4i789R11sMuHLCY27RVI8XloC/GzjRwyhxip5hmIELsHrBq:
```

- b. Vérifiez les clés dhchap du contrôleur :

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

Vous devriez voir une sortie similaire à l'exemple suivant :

```
DHHC-
1:03:jqgYcJSKp73+XqAf2X6twr9ngBpr2n0MGWbmZIZq4PieKZCoilKGef8lAvhYS0P
NK7T+04YD5CRPjh+m3qjJU++yR8s=:
```

Étape 10 : Configurer la sécurité de la couche de transport

Le protocole TLS (Transport Layer Security) assure un chiffrement de bout en bout sécurisé pour les connexions NVMe entre les hôtes NVMe-oF et une baie ONTAP . Vous pouvez configurer TLS 1.3 à l'aide de l'interface de ligne de commande (CLI) et d'une clé pré-partagée (PSK) configurée.



Effectuez les étapes suivantes sur l'hôte SUSE Linux Enterprise Server, sauf indication contraire précisant que vous devez effectuer une étape sur le contrôleur ONTAP .

Étapes

1. Vérifiez que vous disposez des éléments suivants `ktls-utils` , `openssl` , et `libopenssl` paquets installés sur l'hôte :

- a. Vérifiez le `ktls-utils` :

```
rpm -qa | grep ktls
```

Vous devriez voir la sortie suivante affichée :

```
ktls-utils-0.10+33.g311d943-150700.1.5.x86_64
```

- a. Vérifiez les packages SSL :

```
rpm -qa | grep ssl
```

Affiche un exemple de résultat

```
libopenssl3-3.2.3-150700.3.20.x86_64
openssl-3-3.2.3-150700.3.20.x86_64
libopenssl11_1-1.1.1w-150700.9.37.x86_64
```

2. Vérifiez que vous disposez de la configuration correcte pour /etc/tlshd.conf:

```
cat /etc/tlshd.conf
```

Affiche un exemple de résultat

```
[debug]
loglevel=0
tls=0
nl=0
[authenticate]
keyrings=.nvme
[authenticate.client]
#x509.truststore= <pathname>
#x509.certificate= <pathname>
#x509.private_key= <pathname>
[authenticate.server]
#x509.truststore= <pathname>
#x509.certificate= <pathname>
#x509.private_key= <pathname>
```

3. Activer tlshd pour démarrer au démarrage du système :

```
systemctl enable tlshd
```

4. Vérifiez que le tlshd démon est en cours d'exécution :

```
systemctl status tlshd
```

Affiche un exemple de résultat

```
tlshd.service - Handshake service for kernel TLS consumers
   Loaded: loaded (/usr/lib/systemd/system/tlshd.service; enabled;
   preset: disabled)
     Active: active (running) since Wed 2024-08-21 15:46:53 IST; 4h
      57min ago
       Docs: man:tlshd(8)
     Main PID: 961 (tlshd)
        Tasks: 1
         CPU: 46ms
        CGroup: /system.slice/tlshd.service
                  └─961 /usr/sbin/tlshd
Aug 21 15:46:54 RX2530-M4-17-153 tlshd[961]: Built from ktls-utils
0.11-dev on Mar 21 2024 12:00:00
```

5. Générez le TLS PSK en utilisant nvme gen-tls-key :

a. Vérifiez l'hôte :

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

Vous devriez voir le résultat suivant :

```
nqn.2014-08.org.nvmeexpress:uuid:4c4c4544-0035-5910-804b-b2c04f444d33
```

b. Vérifiez la clé :

```
nvme gen-tls-key --hmac=1 --identity=1 --subsysnqn= nqn.1992-
08.com.netapp:sn.a2d41235b78211efb57dd039eab67a95:subsystem.nvme1
```

Vous devriez voir le résultat suivant :

```
NVMeTLSkey-1:01:C50EsaGtuOp8n5fGE9EuWjbBCtshmfoHx4XTqTJUmydf0gIj:
```

6. Sur le contrôleur ONTAP, ajoutez le protocole TLS PSK au sous-système ONTAP :

Affiche un exemple de résultat

```
nvme subsystem host add -vserver vs_iscsi_tcp -subsystem nvme1 -host -nqn nqn.2014-08.org.nvmexpress:uuid:4c4c4544-0035-5910-804b-b2c04f444d33 -tls-configured-psk NVMeTLSkey-1:01:C50EsaGtuOp8n5fGE9EuWjbBCtshmfoHx4XTqTJUmydf0gIj:
```

7. Insérez le TLS PSK dans le porte-clés du noyau hôte :

```
nvme check-tls-key --identity=1 --subsysnqn=nqn.1992-08.com.netapp:sn.a2d41235b78211efb57dd039eab67a95:subsystem.nvme1 --keydata=NVMeTLSkey-1:01:C50EsaGtuOp8n5fGE9EuWjbBCtshmfoHx4XTqTJUmydf0gIj: --insert
```

Vous devriez voir la clé TLS suivante :

```
Inserted TLS key 22152a7e
```



Le PSK montre comme NVMe1R01 parce qu'il utilise identity v1 à partir de l'algorithme de négociation TLS. Identity v1 est la seule version prise en charge par ONTAP.

8. Vérifiez que TLS PSK est correctement inséré :

```
cat /proc/keys | grep NVMe
```

Affiche un exemple de résultat

```
069f56bb I--Q--- 5 perm 3b010000 0 0 psk NVMe1R01
nqn.2014-08.org.nvmexpress:uuid:4c4c4544-0035-5910-804b-b2c04f444d33
nqn.1992-
08.com.netapp:sn.a2d41235b78211efb57dd039eab67a95:subsystem.nvme1
oYVLe1miOwnvDjXKBmrnIgGVpFIBDJtc4hmQXE/36Sw=: 32
```

9. Connectez-vous au sous-système ONTAP à l'aide du protocole TLS PSK inséré :

- Vérifiez le TLS PSK :

```
nvme connect -t tcp -w 192.168.111.80 -a 192.168.111.66 -n nqn.1992-08.com.netapp:sn.a2d41235b78211efb57dd039eab67a95:subsystem.nvme1 --tls_key=0x069f56bb -tls
```

Vous devriez voir le résultat suivant :

```
connecting to device: nvme0
```

a. Vérifiez la liste-sous-système :

```
nvme list-subsy
```

Affiche un exemple de résultat

```
nvme-subsy0 - NQN=nqn.1992-08.com.netapp:sn.a2d41235b78211efb57dd039eab67a95:subsystem.nvme1 hostnqn=nqn.2014-08.org.nvmeexpress:uuid:4c4c4544-0035-5910-804b-b2c04f444d33 \
+- nvme0 tcp
traddr=192.168.111.66, trsvcid=4420, host_traddr=192.168.111.80, src_addr=192.168.111.80 live
```

10. Ajoutez la cible et vérifiez la connexion TLS au sous-système ONTAP spécifié :

```
nvme subsystem controller show -vserver sles15_tls -subsystem sles15 -instance
```

Affiche un exemple de résultat

```
(vserver nvme subsystem controller show)
    Vserver Name: vs_iscsi_tcp
    Subsystem: nvme1
    Controller ID: 0040h
    Logical Interface: tcpnvme_lif1_1
    Node: A400-12-181
    Host NQN: nqn.2014-
08.org.nvmexpress:uuid:4c4c4544-0035-5910-804b-b2c04f444d33
    Transport Protocol: nvme-tcp
    Initiator Transport Address: 192.168.111.80
    Host Identifier:
4c4c454400355910804bb2c04f444d33
    Number of I/O Queues: 2
    I/O Queue Depths: 128, 128
    Admin Queue Depth: 32
    Max I/O Size in Bytes: 1048576
    Keep-Alive Timeout (msec): 5000
    Subsystem UUID: 8bbfb403-1602-11f0-ac2b-
d039eab67a95
    Header Digest Enabled: false
    Data Digest Enabled: false
    Authentication Hash Function: sha-256
    Authentication Diffie-Hellman Group: 3072-bit
    Authentication Mode: unidirectional
    Transport Service Identifier: 4420
    TLS Key Type: configured
    TLS PSK Identity: NVMe1R01 nqn.2014-
08.org.nvmexpress:uuid:4c4c4544-0035-5910-804b-b2c04f444d33
nqn.1992-
08.com.netapp:sn.a2d41235b78211efb57dd039eab67a95:subsystem.nvme1
oYVLelmiOwnvDjXKBmrnIgGVpFIBDJtc4hmQXE/36Sw=
    TLS Cipher: TLS-AES-128-GCM-SHA256
```

Étape 11 : passez en revue les problèmes connus

Il n'y a pas de problème connu.

Informations sur le copyright

Copyright © 2026 NetApp, Inc. Tous droits réservés. Imprimé aux États-Unis. Aucune partie de ce document protégé par copyright ne peut être reproduite sous quelque forme que ce soit ou selon quelque méthode que ce soit (graphique, électronique ou mécanique, notamment par photocopie, enregistrement ou stockage dans un système de récupération électronique) sans l'autorisation écrite préalable du détenteur du droit de copyright.

Les logiciels dérivés des éléments NetApp protégés par copyright sont soumis à la licence et à l'avis de non-responsabilité suivants :

CE LOGICIEL EST FOURNI PAR NETAPP « EN L'ÉTAT » ET SANS GARANTIES EXPRESSES OU TACITES, Y COMPRIS LES GARANTIES TACITES DE QUALITÉ MARCHANDE ET D'ADÉQUATION À UN USAGE PARTICULIER, QUI SONT EXCLUES PAR LES PRÉSENTES. EN AUCUN CAS NETAPP NE SERA TENU POUR RESPONSABLE DE DOMMAGES DIRECTS, INDIRECTS, ACCESSOIRES, PARTICULIERS OU EXEMPLAIRES (Y COMPRIS L'ACHAT DE BIENS ET DE SERVICES DE SUBSTITUTION, LA PERTE DE JOUSSANCE, DE DONNÉES OU DE PROFITS, OU L'INTERRUPTION D'ACTIVITÉ), QUELLES QU'EN SOIENT LA CAUSE ET LA DOCTRINE DE RESPONSABILITÉ, QU'IL S'AGISSE DE RESPONSABILITÉ CONTRACTUELLE, STRICTE OU DÉLICTUELLE (Y COMPRIS LA NÉGLIGENCE OU AUTRE) DÉCOULANT DE L'UTILISATION DE CE LOGICIEL, MÊME SI LA SOCIÉTÉ A ÉTÉ INFORMÉE DE LA POSSIBILITÉ DE TELS DOMMAGES.

NetApp se réserve le droit de modifier les produits décrits dans le présent document à tout moment et sans préavis. NetApp décline toute responsabilité découlant de l'utilisation des produits décrits dans le présent document, sauf accord explicite écrit de NetApp. L'utilisation ou l'achat de ce produit ne concède pas de licence dans le cadre de droits de brevet, de droits de marque commerciale ou de tout autre droit de propriété intellectuelle de NetApp.

Le produit décrit dans ce manuel peut être protégé par un ou plusieurs brevets américains, étrangers ou par une demande en attente.

LÉGENDE DE RESTRICTION DES DROITS : L'utilisation, la duplication ou la divulgation par le gouvernement sont sujettes aux restrictions énoncées dans le sous-paragraphe (b)(3) de la clause Rights in Technical Data-Noncommercial Items du DFARS 252.227-7013 (février 2014) et du FAR 52.227-19 (décembre 2007).

Les données contenues dans les présentes se rapportent à un produit et/ou service commercial (tel que défini par la clause FAR 2.101). Il s'agit de données propriétaires de NetApp, Inc. Toutes les données techniques et tous les logiciels fournis par NetApp en vertu du présent Accord sont à caractère commercial et ont été exclusivement développés à l'aide de fonds privés. Le gouvernement des États-Unis dispose d'une licence limitée irrévocable, non exclusive, non cessible, non transférable et mondiale. Cette licence lui permet d'utiliser uniquement les données relatives au contrat du gouvernement des États-Unis d'après lequel les données lui ont été fournies ou celles qui sont nécessaires à son exécution. Sauf dispositions contraires énoncées dans les présentes, l'utilisation, la divulgation, la reproduction, la modification, l'exécution, l'affichage des données sont interdits sans avoir obtenu le consentement écrit préalable de NetApp, Inc. Les droits de licences du Département de la Défense du gouvernement des États-Unis se limitent aux droits identifiés par la clause 252.227-7015(b) du DFARS (février 2014).

Informations sur les marques commerciales

NETAPP, le logo NETAPP et les marques citées sur le site <http://www.netapp.com/TM> sont des marques déposées ou des marques commerciales de NetApp, Inc. Les autres noms de marques et de produits sont des marques commerciales de leurs propriétaires respectifs.